


```

      AAAAAA      EEEEEEEEE EEEEEEEEE DDDDDDDD MM MM AAAAAA I I I I I I NN NN
      AAAAAA      EEEEEEEEE EEEEEEEEE DDDDDDDD MM MM AAAAAA I I I I I I NN NN
AA      AA      EE      DD      DD      MMMM MMMM AA      AA      NN NN NN
AA      AA      EE      DD      DD      MMMM MMMM AA      AA      NN NN NN
AA      AA      EE      DD      DD      MM MM MM AA      AA      NNNN NN
AA      AA      EE      DD      DD      MM MM MM AA      AA      NNNN NN
AA      AA      EEEEEEEE DD      DD      MM MM MM AA      AA      NN NN NN
AA      AA      EEEEEEEE DD      DD      MM MM MM AAAAAAAAAA I I I I I I NN NN
AAAAAAAAAA EE      DD      DD      MM MM MM AAAAAAAAAA I I I I I I NN NN
AAAAAAAAAA EE      DD      DD      MM MM MM AAAAAAAAAA I I I I I I NN NN
AA      AA      EE      DD      DD      MM MM MM AA      AA      NN NN NN
AA      AA      EEEEEEEEE DD      DD      MM MM MM AA      AA      NN NN NN
AA      AA      EEEEEEEEE DDDDDDDD DD      MM MM MM AA      AA      NN NN NN
AA      AA      EEEEEEEEE DDDDDDDD DD      MM MM MM AA      AA      NN NN NN

```

```

LL      I I I I I I SSSSSSSS
LL      I I I I I I SSSSSSSS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SSSSSS
LL      I I      SSSSSS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LLLLLLLLLL I I I I I I SSSSSSSS
LLLLLLLLLL I I I I I I SSSSSSSS

```

```
1 0001 0 MODULE AED$MAIN (
2 0002 0 LANGUAGE (BLISS32),
3 0003 0 IDENT = 'V04-000'
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1 ++
31 0031 1
32 0032 1 FACILITY: Miscellaneous utilities
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 This module contains the routines for processing the user's input and
37 0037 1 updating the object's ACL in the appropriate manner.
38 0038 1
39 0039 1 ENVIRONMENT:
40 0040 1
41 0041 1 VAX/VMS operating system, user mode utilities.
42 0042 1
43 0043 1 --
44 0044 1
45 0045 1
46 0046 1 AUTHOR: L. Mark Pilant CREATION DATE: 12-Nov-1982 9:50
47 0047 1
48 0048 1 MODIFIED BY:
49 0049 1
50 0050 1 V03-016 LMP0291 L. Mark Pilant, 31-Jul-1984 13:15
51 0051 1 Correct a bug that caused the editor to loop forever in the
52 0052 1 ACE after the one being deleted was more than one line.
53 0053 1
54 0054 1 V03-015 LMP0268 L. Mark Pilant, 28-Jun-1984 15:01
55 0055 1 Don't explicitly save the journal file on a QUIT.
56 0056 1
57 0057 1 V03-014 LMP0267 L. Mark Pilant, 28-Jun-1984 12:15
```



```

58      0058 1
59      0059 1
60      0060 1
61      0061 1
62      0062 1
63      0063 1
64      0064 1
65      0065 1
66      0066 1
67      0067 1
68      0068 1
69      0069 1
70      0070 1
71      0071 1
72      0072 1
73      0073 1
74      0074 1
75      0075 1
76      0076 1
77      0077 1
78      0078 1
79      0079 1
80      0080 1
81      0081 1
82      0082 1
83      0083 1
84      0084 1
85      0085 1
86      0086 1
87      0087 1
88      0088 1
89      0089 1
90      0090 1
91      0091 1
92      0092 1
93      0093 1
94      0094 1
95      0095 1
96      0096 1
97      0097 1
98      0098 1
99      0099 1
100     0100 1
101     0101 1
102     0102 1
103     0103 1
104     0104 1
105     0105 1
106     0106 1
107     0107 1
108     0108 1
109     0109 1
110     0110 1
111     0111 1

Add support for an ADVANCE FIELD key.

V03-013 LMP0250      L. Mark Pilant,      4-May-1984 15:42
Fix a bug introduced by LMP0238 that caused the wrong item
code to be used when updating ACLs.

V03-012 LMP0238      L. Mark Pilant,      19-Apr-1984 13:32
Use the size of the ACE for twiddling, when possible.

V03-011 LMP0230      L. Mark Pilant,      16-Apr-1984 9:25
Track changes made to the $CHANGE_ACL system service.

V03-010 LMP0213      L. Mark Pilant,      24-Mar-1984 12:23
Add support for locking and unlocking the object's ACL.

V03-009 LMP0193      L. Mark Pilant,      15-Feb-1984 9:59
Add support for additional editor actions: delete to EOL,
reset, and quit. Also move the actual ACL updating to the
session termination routine.

V03-008 LMP0172      L. Mark Pilant,      28-Nov-1983 12:11
Numerous bug fixes, support for VT2xx terminals, and a
session keystroke logger.

V03-007 LMP0161      L. Mark Pilant,      5-Oct-1983 10:36
Make sure that the modified ACE gets written out when
crossing ACE boundaries during a search operation.

V03-006 LMP0147      L. Mark Pilant,      29-Aug-1983 12:46
Fix a bug that caused the display to be incorrect when
un-deleting an ACE as the last line. Also fix a bug that
caused te display to be wrong when a ^U is given in the
middle of a line.

V03-005 LMP0138      L. Mark Pilant,      16-Aug-1983 13:23
Misc fixes to prompting mode input.

V03-004 LMP0103      L. Mark Pilant,      20-Apr-1983 11:23
Add support for HIDDEN ACEs. Also misc fixes to prompting.

V03-003 LMP0081      L. Mark Pilant,      16-Feb-1983 10:20
Correct some minor bugs with the string searching routines.

V03-002 LMP0076      L. Mark Pilant,      1-Feb-1983 13:07
Add support for a key definition file.

V03-001 LMP0074      L. Mark Pilant,      21-Jan-1983 16:54
Random fixes and support for RMS journaling ACE's.

**

LIBRARY 'SYSS$LIBRARY:LIB.L32';
LIBRARY 'SYSS$LIBRARY:TPAMAC.L32';
REQUIRE 'SRC$:ACLEDTDEF';
```



```
113 0564 1 FORWARD ROUTINE
114 0565 1     AED_PROCESSACL : NOVALUE,           ! Main processing routine
115 0566 1
116 0567 1 ! The following are routines called based upon the editor action desired.
117 0568 1
118 0569 1     ACT_RUB_CHR,
119 0570 1     ACT_RUB_WRD,
120 0571 1     ACT_RUB_BOL,
121 0572 1     ACT_DEL_CHR,
122 0573 1     ACT_DEL_WRD,
123 0574 1     ACT_DEL_EOL,
124 0575 1     ACT_DEL_ACE,
125 0576 1     ACT_UNDEL_CHR,
126 0577 1     ACT_UNDEL_WRD,
127 0578 1     ACT_UNDEL_LIN,
128 0579 1     ACT_UNDEL_ACE,
129 0580 1     ACT_MOVE_WRD,
130 0581 1     ACT_MOVE_ACE,
131 0582 1     ACT_MOVE_BOL,
132 0583 1     ACT_MOVE_EOL,
133 0584 1     ACT_UP,
134 0585 1     ACT_DOWN,
135 0586 1     ACT_RIGHT,
136 0587 1     ACT_LEFT,
137 0588 1     ACT_TOP,
138 0589 1     ACT_BOTTOM,
139 0590 1     ACT_FIND_STR,
140 0591 1     ACT_FIND_NXT,
141 0592 1     ACT_ADV_FIELD,
142 0593 1     ACT_SEL_FIELD,
143 0594 1     ACT_SEL_ITEM,
144 0595 1     ACT_HELP,
145 0596 1     ACT_REFRESH,
146 0597 1     ACT_ENTER,
147 0598 1     ACT_INSERT,
148 0599 1     ACT_EXIT,
149 0600 1
150 0601 1 ! The following are common ACE text manipulating routines.
151 0602 1
152 0603 1     FINISH_ACE      : NOVALUE;           ! Tie off the ACE
153 0604 1
154 0605 1 EXTERNAL ROUTINE
155 0606 1     AED_PUTOUTPUT,      ! General purpose output routine
156 0607 1     AED_GIVEHELP,      ! Interactive help routine
157 0608 1     AED_UPDATEACL,      ! Update file's ACL
158 0609 1     AED_SET_CURSOR,      ! Set cursor position
159 0610 1     AED_SELECTFIELD : NOVALUE,      ! Select next ACE field
160 0611 1     AED_SELECTITEM  : NOVALUE,      ! Select next ACE item
161 0612 1     AED_SETACETYPE  : NOVALUE,      ! Set ACE type text
162 0613 1     AED_COMPRESS   : NOVALUE,      ! Compress the display
163 0614 1     AED_POSITION   : NOVALUE,      ! Position to selected line
164 0615 1     AED_COPYSEGMENT,    ! Copy segment to working storage
165 0616 1     AED_REPSEGMENT,    ! Replace with working storage segment
166 0617 1     AED_SEGSPLIT,      ! Split segment into two pieces
167 0618 1     AED_SEGCOMBINE,    ! Combine two line segments
168 0619 1     AED_DECODEKEY,      ! Key action decoder
169 0620 1
```

```
: 170      0621 1 ! Macros to make working with line segments easier.
: 171      0622 1
: 172      0623 1 MACRO
: 173      0624 1     BUFFER_CHAR = INPUT_BUFFER[.BUFFER_INDEX] %;
: 174      0625 1
: 175      0626 1 ! Storage used by all the routines in this module.
: 176      0627 1
: 177      0628 1 OWN
: 178      0629 1     BUFFER_INDEX,      : $BBLOCK [DSC$C_S_BLN], ! Index into input storage
: 179      0630 1     ECHO_DESC          : ! Text echoing descr
: 180      0631 1     TEMP_LINE          : Temp copy of line number
: 181      0632 1     REMOVED_LINE        : REF $BBLOCK,      ! Address of line removed
: 182      0633 1     REMOVED_ACE        : REF $BBLOCK,      ! Address of ACE removed
: 183      0634 1     NEW_TEXT_LINE      : REF $BBLOCK,      ! Address of new line storage
: 184      0635 1     CHAR_PROCESSED,    : Chars checked by ACL parser
: 185      0636 1     APPEND_INDEX,      : Index for combining segments
: 186      0637 1     DUMMY_LINE         : REF $BBLOCK,      ! Temp line pointer
: 187      0638 1     TERM_CHAR          : VECTOR [1,BYTE],  ! Character/code input
: 188      0639 1     SEARCH_SIZE        : VECTOR [1,WORD],  ! Search string size
: 189      0640 1     SEARCH_STRING      : VECTOR [512,BYTE]; ! Search string buffer
: 190      0641 1
: 191      0642 1 BIND
: 192      0643 1     SEGMENT_SIZE        = AED_T_CURLINE[LINE_W_SIZE] : WORD,
: 193      0644 1     ! Input line segment size
: 194      0645 1     INPUT_BUFFER        = AED_T_CURLINE[LINE_T_TEXT] : VECTOR [,BYTE];
: 195      0646 1     ! Input line segment text
```



```
197 0647 1 ZSBTTL 'AED_PROCESSACL - main processing loop'
198 0648 1 GLOBAL ROUTINE AED_PROCESSACL : NOVALUE =
199 0649 1
200 0650 1 ++
201 0651 1
202 0652 1 FUNCTIONAL DESCRIPTION:
203 0653 1
204 0654 1     This routine is the main processing loop for the ACL editor. It
205 0655 1     accepts the users input, which may be a new ACE or modifications
206 0656 1     to an existing ACE, and updates the in core ACL as appropriate.
207 0657 1
208 0658 1 CALLING SEQUENCE:
209 0659 1     AED_PROCESSACL ()
210 0660 1
211 0661 1 INPUT PARAMETERS:
212 0662 1     none
213 0663 1
214 0664 1 IMPLICIT INPUTS:
215 0665 1     AED_W_TERMIN: terminal input channel
216 0666 1     AED_Q_LINETABLE: input line text queue
217 0667 1     AED_L_CURACE: address of current ACE
218 0668 1
219 0669 1 OUTPUT PARAMETERS:
220 0670 1     none
221 0671 1
222 0672 1 IMPLICIT OUTPUTS:
223 0673 1     AED_L_FIRSTLINE: address of first line segment of ACE
224 0674 1     AED_L_LASTLINE: address of last line segment of ACE
225 0675 1
226 0676 1 ROUTINE VALUE:
227 0677 1     none
228 0678 1
229 0679 1 SIDE EFFECTS:
230 0680 1     The object's ACL is appropriately modified.
231 0681 1
232 0682 1 --
233 0683 1
234 0684 2 BEGIN
235 0685 2
236 0686 2 LABEL
237 0687 2     INPUT;                                ! User input loop
238 0688 2
239 0689 2 LOCAL
240 0690 2     LOCAL_STATUS,                        ! Local routine exit status
241 0691 2     SPLIT_SEGMENT : REF $BBLOCK,          ! Pointer to remaining text
242 0692 2     SPLIT_SIZE;                          ! Size of remaining text
243 0693 2
244 0694 2 ! Initialize all variables and flags.
245 0695 2
246 0696 2 CH$FILL (0, DSC$C_S_BLN, ECHO_DESC);
247 0697 2 CH$MOVE (DSC$C_S_BLN, ECHO_DESC, AED_Q_DEL_WORD);
248 0698 2 CH$MOVE (DSC$C_S_BLN, ECHO_DESC, AED_Q_DEL_LINE);
249 0699 2 AED_B_DEL_CHAR = 0;
250 0700 2 BUFFER_INDEX = 0;
251 0701 2
252 0702 2 ! Set up initial display variables.
253 0703 2
```



```
254 0704 2 AED_Q_DEL_ACE[LINE_L_FLINK] = AED_Q_DEL_ACE[LINE_L_FLINK];
255 0705 2 AED_Q_DEL_ACE[LINE_L_BLINK] = AED_Q_DEL_ACE[LINE_L_FLINK];
256 0706 2
257 0707 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 1;
258 0708 2
259 0709 2 AED_L_FLAGS[AED_V_ACERROR] = 1; ! Clear message area
260 0710 2
261 0711 2 ! If there is no ACL (the display is empty), set up to append the text
262 0712 2 ! entered. Otherwise, set up to modify the first segment of the display.
263 0713 2
264 0714 2 IF .AED_Q_LINETABLE[LINE_L_FLINK] EQLA AED_Q_LINETABLE[LINE_L_FLINK]
265 0715 2 THEN
266 0716 2 BEGIN
267 0717 2 AED_L_FLAGS[AED_V_ENDACL] = 1; ! At the end of the ACL
268 0718 2 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
269 0719 2 AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
270 0720 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_Q_LINETABLE[LINE_L_BLINK]);
271 0721 2 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
272 0722 2 AED_L_FIRSTLINE[LINE_Q_FLAGS] = LINE_M_BEGINACE;
273 0723 2 AED_L_CURACE = 0;
274 0724 2 IF .AED_L_FLAGS[AED_V_PROMPT]
275 0725 2 THEN
276 0726 2 BEGIN
277 0727 2 AED_B_ACETYPE = 0;
278 0728 2 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
279 0729 2 AED_SELECTFIELD (BUFFER_INDEX);
280 0730 2 ECHO_DESC[DESCW_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
281 0731 2 ECHO_DESC[DESC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
282 0732 2 SCR$SET CURSOR T.AED_B_LINE, 1;
283 0733 2 AED_PUTOUTPUT (ECHO_DESC);
284 0734 2 SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
285 0735 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
286 0736 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
287 0737 2 END;
288 0738 2 END
289 0739 2 ELSE
290 0740 2 BEGIN
291 0741 2 AED_COPSEGMENT (.AED_Q_LINETABLE[LINE_L_FLINK]);
292 0742 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], AED_Q_LINETABLE[LINE_L_FLINK]);
293 0743 2 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
294 0744 2 AED_W_TOTALSIZE = .AED_L_FIRSTLINE[LINE_Q_SIZE];
295 0745 2 UNTIL .AED_L_LASTLINE[LINE_V_ENDACE]
296 0746 2 DO
297 0747 2 BEGIN
298 0748 2 IF .AED_L_LASTLINE EQLA AED_T_CURLINE
299 0749 2 THEN AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
300 0750 2 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
301 0751 2 AED_W_TOTALSIZE = .AED_Q_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
302 0752 2 END;
303 0753 2 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
304 0754 2 IF .AED_L_FLAGS[AED_V_PROMPT]
305 0755 2 THEN
306 0756 2 BEGIN
307 0757 2 AED_L_FLAGS[AED_V_NOITEMSEL] = 1;
308 0758 2 AED_SELECTFIELD (BUFFER_INDEX);
309 0759 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
310 0760 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
```

```
311 0761 3      END;
312 0762 3      END;
313 0763 3      AED_L_BEGINLINE = .AED_Q_LINETABLE[LINE_L_FLINK];
314 0764 3
315 0765 3      ! Loop getting characters from the user until an End-Of-file is seen.  When
316 0766 3      ! an EOF is seen, it indicates the end of the session.
317 0767 3
318 0768 3      WHILE 1
319 0769 3      DO
320 0770 3          INPUT: BEGIN
321 0771 3          TERM_CHAR = AED_DECODEKEY ();
322 0772 3          IF .TERM_CHAR EOL 0 THEN RETURN;
323 0773 3          IF .AED_C_FLAGS[AED_V_ACERROR] AND .AED_L_FLAGS[AED_V_SCOPE]
324 0774 3          THEN
325 0775 3              BEGIN
326 0776 3              SCRSErase PAGE (21, 1);
327 0777 3              AED_SET_CURSOR (.AED_B_LINE, .BUFFER_INDEX + 1);
328 0778 3              AED_L_FLAGS[AED_V_ACERROR] = 0;
329 0779 3              END;
330 0780 3
331 0781 3      ! Choose the appropriate action based upon the character typed.
332 0782 3
333 0783 3      WHILE .AED_L_FLAGS[AED_V_ACTIONKEY]
334 0784 3      DO
335 0785 3          BEGIN
336 0786 3          CASE .TERM_CHAR FROM 1 TO KEY_C_MAX_CODE-1 OF
337 0787 3          SET
338 0788 3
339 0789 3      ! Actions to delete text.
340 0790 3
341 0791 3          [KEY_C_RUB_CHR]:      ACT_RUB_CHR ();
342 0792 3
343 0793 3          [KEY_C_RUB_WRD]:
344 0794 3              BEGIN
345 0795 3              LOCAL STATUS = ACT_RUB_WRD ();
346 0796 3              IF NOT .LOCAL_STATUS THEN RETURN;
347 0797 3              END;
348 0798 3
349 0799 3          [KEY_C_RUB_BOL]:      ACT_RUB_BOL ();
350 0800 3
351 0801 3          [KEY_C_DEL_CHR]:      ACT_DEL_CHR ();
352 0802 3
353 0803 3          [KEY_C_DEL_WRD]:
354 0804 3              BEGIN
355 0805 3              LOCAL STATUS = ACT_DEL_WRD ();
356 0806 3              IF NOT .LOCAL_STATUS THEN RETURN;
357 0807 3              END;
358 0808 3
359 0809 3          [KEY_C_DEL_EOL]:
360 0810 3              BEGIN
361 0811 3              LOCAL STATUS = ACT_DEL_EOL ();
362 0812 3              IF NOT .LOCAL_STATUS THEN RETURN;
363 0813 3              END;
364 0814 3
365 0815 3          [KEY_C_DEL_ACE]:
366 0816 3              BEGIN
367 0817 3              LOCAL STATUS = ACT_DEL_ACE ();
```



```
368 0818 5      IF NOT .LOCAL_STATUS THEN RETURN;
369 0819 4      END;
370 0820 4
371 0821 4 ! Actions to restore deleted text.
372 0822 4
373 0823 4      [KEY_C_UNDEL_CHR]:      ACT_UNDEL_CHR ();
374 0824 4
375 0825 4      [KEY_C_UNDEL_WRD]:      ACT_UNDEL_WRD ();
376 0826 4
377 0827 4      [KEY_C_UNDEL_LIN]:      ACT_UNDEL_LIN ();
378 0828 4
379 0829 4      [KEY_C_UNDEL_ACE]:
380 0830 5      BEGIN
381 0831 5          LOCAL_STATUS = ACT_UNDEL_ACE ();
382 0832 5          IF NOT .LOCAL_STATUS THEN RETURN;
383 0833 4      END;
384 0834 4
385 0835 4 ! Actions to move through the ACL independant of the direction.
386 0836 4
387 0837 4      [KEY_C_UP]:
388 0838 5      BEGIN
389 0839 5          LOCAL_STATUS = ACT_UP ();
390 0840 5          IF NOT .LOCAL_STATUS THEN RETURN;
391 0841 4      END;
392 0842 4
393 0843 4      [KEY_C_DOWN]:
394 0844 5      BEGIN
395 0845 5          LOCAL_STATUS = ACT_DOWN ();
396 0846 5          IF NOT .LOCAL_STATUS THEN RETURN;
397 0847 4      END;
398 0848 4
399 0849 4      [KEY_C_RIGHT]: ACT_RIGHT ();
400 0850 4
401 0851 4      [KEY_C_LEFT]:  ACT_LEFT ();
402 0852 4
403 0853 4      [KEY_C_TOP]:
404 0854 5      BEGIN
405 0855 5          LOCAL_STATUS = ACT_TOP ();
406 0856 5          IF NOT .LOCAL_STATUS THEN RETURN;
407 0857 4      END;
408 0858 4
409 0859 4      [KEY_C_BOTTOM]:
410 0860 5      BEGIN
411 0861 5          LOCAL_STATUS = ACT_BOTTOM ();
412 0862 5          IF NOT .LOCAL_STATUS THEN RETURN;
413 0863 4      END;
414 0864 4
415 0865 4 ! Set the direction of the move.
416 0866 4
417 0867 4      [KEY_C_ADVANCE]:
418 0868 5      BEGIN
419 0869 5          AED_L_FLAGS[AED_V_BACKWARD] = 0;
420 0870 5          AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
421 0871 5          TERM_CHAR = 0;
422 0872 4      END;
423 0873 4
424 0874 4      [KEY_C_BACKUP]:
```


AED_PROCESSACL - main processing loop

```
425 0875 S BEGIN
426 0876 S AED_L_FLAGS[AED_V_BACKWARD] = 1;
427 0877 S AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
428 0878 S TERM_CHAR = 0;
429 0879 S END;
430 0880 S
431 0881 S ! Advance through the ACL based upon the direction chosen.
432 0882 S
433 0883 S [KEY_C_MOVE_WRD]: ACT_MOVE_WRD ();
434 0884 S
435 0885 S [KEY_C_MOVE_BOL]: ACT_MOVE_BOL ();
436 0886 S
437 0887 S [KEY_C_MOVE_EOL]: ACT_MOVE_EOL ();
438 0888 S
439 0889 S [KEY_C_MOVE_ACE]:
440 0890 S BEGIN
441 0891 S LOCAL STATUS = ACT_MOVE_ACE ();
442 0892 S IF NOT .LOCAL_STATOS THEN RETURN;
443 0893 S END;
444 0894 S
445 0895 S [KEY_C_FIND_STR]:
446 0896 S BEGIN
447 0897 S LOCAL STATUS = ACT_FIND_STR ();
448 0898 S IF NOT .LOCAL_STATOS THEN RETURN;
449 0899 S END;
450 0900 S
451 0901 S [KEY_C_FIND_NXT]:
452 0902 S BEGIN
453 0903 S LOCAL STATUS = ACT_FIND_NXT ();
454 0904 S IF NOT .LOCAL_STATOS THEN RETURN;
455 0905 S END;
456 0906 S
457 0907 S ! Advance through an ACE using fields and items.
458 0908 S
459 0909 S [KEY_C_ADV_FIELD]:
460 0910 S BEGIN
461 0911 S LOCAL STATUS = ACT_ADV_FIELD ();
462 0912 S IF NOT .LOCAL_STATOS THEN RETURN;
463 0913 S END;
464 0914 S
465 0915 S [KEY_C_SEL_FIELD]:
466 0916 S BEGIN
467 0917 S LOCAL STATUS = ACT_SEL_FIELD ();
468 0918 S IF NOT .LOCAL_STATOS THEN RETURN;
469 0919 S END;
470 0920 S
471 0921 S [KEY_C_SEL_ITEM]:
472 0922 S BEGIN
473 0923 S LOCAL STATUS = ACT_SEL_ITEM ();
474 0924 S IF NOT .LOCAL_STATOS THEN RETURN;
475 0925 S END;
476 0926 S
477 0927 S ! Miscellaneous editor actions.
478 0928 S
479 0929 S [KEY_C_GOLD]:
480 0930 S BEGIN
481 0931 S AED_L_FLAGS[AED_V_GOLDKEY] = 1;
```

```
482 0932 S      AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
483 0933 S      TERM_CHAR = 0;
484 0934 S      END;
485 0935 S
486 0936 S      [KEY C HELP]:
487 0937 S      BEGIN
488 0938 S      AED_L_FLAGS[AED_V_ACEFORMAT] = 0;
489 0939 S      ACT_HELP ();
490 0940 S      END;
491 0941 S
492 0942 S      [KEY C HELPFMT]:
493 0943 S      BEGIN
494 0944 S      AED_L_FLAGS[AED_V_ACEFORMAT] = 1;
495 0945 S      ACT_HELP ();
496 0946 S      END;
497 0947 S
498 0948 S      [KEY C ENTER]:
499 0949 S      BEGIN
500 0950 S      LOCAL STATUS = ACT_ENTER ();
501 0951 S      IF NOT .LOCAL_STATUS THEN RETURN;
502 0952 S      END;
503 0953 S
504 0954 S      [KEY C INSERT]:
505 0955 S      BEGIN
506 0956 S      LOCAL STATUS = ACT_INSERT ();
507 0957 S      IF NOT .LOCAL_STATUS THEN RETURN;
508 0958 S      END;
509 0959 S
510 0960 S      [KEY_C_REFRESH]:      ACT_REFRESH (0);
511 0961 S
512 0962 S      [KEY_C_RESET]:      ACT_REFRESH (1);
513 0963 S
514 0964 S      [KEY C EXIT]:
515 0965 S      BEGIN
516 0966 S      LOCAL STATUS = ACT_EXIT (0);
517 0967 S      IF NOT .LOCAL_STATUS THEN RETURN;
518 0968 S      END;
519 0969 S
520 0970 S      [KEY C QUIT]:
521 0971 S      BEGIN
522 0972 S      ACT_EXIT (1);
523 0973 S      RETURN;
524 0974 S      END;
525 0975 S
526 0976 S      [KEY C OVERSTRIKE]:
527 0977 S      BEGIN
528 0978 S      AED_L_FLAGS[AED_V_OVERSTRIKE] = NOT .AED_L_FLAGS[AED_V_OVERSTRIKE];
529 0979 S      LEAVE INPUT;
530 0980 S      END;
531 0981 S
532 0982 S      [KEY C DEBUG]:
533 0983 S      BEGIN
534 0984 S      LOCAL      PREV_HANDLER;
535 0985 S      EXTERNAL ROUTINE LIB$SIGNAL : ADDRESSING_MODE (GENERAL);
536 0986 S
537 0987 S      ! If the debugger is not present, this is a no-op.
538 0988 S
```

AED_PROCESSACL - main processing loop

```
0989      $SETEXV (VECTOR = 0, PRVHND = PREV_HANDLER);
0990      IF .PREV_HANDLER EQL 0 THEN LEAVE INPUT;
0991      $SETEXV (VECTOR = 0, ADDRES = .PREV_HANDLER);
0992
0993      ! Enter the debugger.
0994
0995      SCR$SET CURSOR (21, 1);
0996      LIB$SIGNAL (SS$ DEBUG);
0997      SCR$SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
0998      LEAVE INPUT;
0999      END;
1000
1001      ! End of the ACTIONKEY case statement.
1002
1003      [INRANGE,OUTRANGE]:      LEAVE INPUT;
1004
1005      YES:
1006      END;
1007
1008      ! See if it is necessary to continue.
1009
1010      IF .TERM_CHAR EQL 0 THEN LEAVE INPUT;
1011
1012      ! If the current ACE is marked as untouchable, no modifications are allowed.
1013
1014      IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1015      THEN
1016      BEGIN
1017      SIGNAL (AED$_NOMODIFY);
1018      LEAVE INPUT;
1019      END;
1020
1021      ! Carriage return - terminate current line segment
1022
1023      IF NOT .AED_L_FLAGS[AED_V_ACTIONKEY]
1024      THEN SELECT ONE .TERM_CHAR OF
1025      SET
1026      [X'0D']:
1027      BEGIN
1028
1029      ! Tie off the end of the current segment.
1030
1031      AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1032      IF .AED_L_FLAGS[AED_V_PROMPT]
1033      AND .BUFFER_INDEX GEQ .SEGMENT_SIZE
1034      THEN
1035      BEGIN
1036      IF .INPUT_BUFFER[.BUFFER_INDEX - 1] NEQ ' '
1037      AND .INPUT_BUFFER[.BUFFER_INDEX - 1] NEQ '='
1038      AND .INPUT_BUFFER[.BUFFER_INDEX - 1] NEQ ')'
1039      AND NOT .AED_L_FLAGS[AED_V_OPENUI]
1040      AND .AED_B_FIELD LSS 2
1041      AND .SEGMENT_SIZE GTR 0
1042      THEN
1043      BEGIN
1044      BUFFER_CHAR = ' ';
1045      ECHO_DESC[DESC$W_LENGTH] = 1;
```



```
596      1046 6      ECHO DESC[DSC$A POINTER] = BUFFER_CHAR;
597      1047 6      AED_PUTOUTPUT (ECHO_DESC);
598      1048 6      AED_B_COLUMN = .AED_B_COLUMN + 1;
599      1049 6      BUFFER_INDEX = .BUFFER_INDEX + 1;
600      1050 6      SEGMENT_SIZE = .SEGMENT_SIZE + 1;
601      1051 5      END;
602      1052 4      END;
603      1053 4
604      1054 4      ! Split the line.
605      1055 4
606      1056 4      NEW_TEXT_LINE = AED_SEGSPLIT (BUFFER_INDEX, 1, 0, 0);
607      1057 4
608      1058 4      ! See if a new prompt string is necessary.
609      1059 4
610      1060 4      IF .AED_L_FLAGS[AED_V_PROMPT]
611      1061 4      AND NOT .AED_L_FLAGS[AED_V_OPENUIIC]
612      1062 4      AND NOT .AED_L_FLAGS[AED_V_NOITEMSEL]
613      1063 4      AND .AED_L_LASTLINE EQL .AED_T_CURLINE[LINE_L_FLINK]
614      1064 4      AND .AED_L_FIRSTLINE[LINE_L_BIRACE] EQL 0
615      1065 4      THEN
616      1066 4      BEGIN
617      1067 4      AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
618      1068 4      AED_SELECTFIELD (BUFFER_INDEX);
619      1069 4      ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
620      1070 4      ECHO_DESC[DSC$A_POINTER] = .AED_T_CURLINE[LINE_T_TEXT];
621      1071 4      SCR$SET CURSOR (.AED_B_LINE, 1);
622      1072 4      AED_PUTOUTPUT (ECHO_DESC);
623      1073 4      SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
624      1074 4      AED_B_COLUMN = .BUFFER_INDEX + 1;
625      1075 4      END;
626      1076 4      AED_SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
627      1077 4      AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
628      1078 4      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
629      1079 4      LEAVE INPUT;
630      1080 4      END;
631      1081 4
632      1082 4      ! All other characters
633      1083 4
634      1084 4      [OTHERWISE]:
635      1085 4      BEGIN
636      1086 4
637      1087 4      ! Check for special characters.
638      1088 4
639      1089 4      IF .TERM_CHAR LSS ' ' THEN LEAVE INPUT;      ! Ignore control chars
640      1090 4      AED_L_FLAGS[AED_V_MODIFIED] = 1;
641      1091 4      AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
642      1092 4      IF .TERM_CHAR GEQ 'a' AND .TERM_CHAR LEQ 'z'
643      1093 4      OR .TERM_CHAR GEQ 'X'EO' AND .TERM_CHAR LEQ 'X'FE'
644      1094 4      THEN TERM_CHAR = .TERM_CHAR - 32;      ! Lower to upper case letters
645      1095 4
646      1096 4      ! Echo the character just typed at the current position or split the line and
647      1097 4      ! echo the character.
648      1098 4
649      1099 4      IF (.BUFFER_INDEX GEQ .AED_L_PAGEWIDTH)
650      1100 4      OR (NOT .AED_L_FLAGS[AED_V_OVERSTRIKE]
651      1101 4      AND .SEGMENT_SIZE GEQ .AED_L_PAGEWIDTH)
652      1102 4      THEN AED_SEGSPLIT (BUFFER_INDEX, 0, 0, 0);
```

```
653 1103 4
654 1104 4      ECHO_DESC[DSCSW_LENGTH] = 1;
655 1105 4      ECHO_DESC[DSCSA_POINTER] = TERM_CHAR;
656 1106 4      AED_PUTOUTPUT (ECHO_DESC);
657 1107 4
658 1108 4      ! If the character was entered in insert mode, move all of the characters
659 1109 4      ! over one position.
660 1110 4
661 1111 4      IF .BUFFER_INDEX LSS .SEGMENT_SIZE
662 1112 4      AND NOT .AED_L_FLAGS[AED_V_OVERSTRIKE]
663 1113 4      THEN
664 1114 4          BEGIN
665 1115 4              ECHO_DESC[DSCSW_LENGTH] = .SEGMENT_SIZE - .BUFFER_INDEX;
666 1116 4              ECHO_DESC[DSCSA_POINTER] = BUFFER_CHAR;
667 1117 4              AED_PUTOUTPUT (ECHO_DESC);
668 1118 4              AED_SET_CURSOR (.AED_B_LINE, .BUFFER_INDEX + 2);
669 1119 4              CHSCOPY (.ECHO_DESC[DSCSW_LENGTH], BUFFER_CHAR,
670 1120 4                  0,
671 1121 4                  512 - .BUFFER_INDEX - 1, INPUT_BUFFER[.BUFFER_INDEX + 1]);
672 1122 4          END;
673 1123 4
674 1124 4      IF .TERM_CHAR EQL '[' THEN AED_L_FLAGS[AED_V_OPENUIIC] = 1;
675 1125 4      IF .TERM_CHAR EQL ']' THEN AED_L_FLAGS[AED_V_OPENUIIC] = 0;
676 1126 4
677 1127 4      ! Now put the entered character into the line buffer.
678 1128 4
679 1129 4      BUFFER_CHAR = .TERM_CHAR;
680 1130 4      BUFFER_INDEX = .BUFFER_INDEX + 1;
681 1131 4      AED_B_COLUMN = .BUFFER_INDEX + 1;
682 1132 4
683 1133 4      ! If in insert mode, the segment size has grown by one character.
684 1134 4
685 1135 4      IF NOT .AED_L_FLAGS[AED_V_OVERSTRIKE]
686 1136 4      OR .BUFFER_INDEX GEQ .SEGMENT_SIZE
687 1137 4      THEN SEGMENT_SIZE = .SEGMENT_SIZE + 1;
688 1138 4      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
689 1139 4      END;
690 1140 3      TES;
691 1141 2      END;
692 1142 2
693 1143 2      RETURN;
694 1144 1      END;
```

! End of routine AED_PROCESSACL

.TITLE AED\$MAIN
.IDENT \V04-000\

.PSECT AED_COMMON,NOEXE, OVR,0

00000 AED_L_FLAGS:
 .BLK8 4
00004 AED_B_OPTIONS:
 .BLK8 1
00005 .BLK8 3
00008 AED_L_OBJTYP:
 .BLK8 4
0000C AED_Q_OBJNAM:

00014	AED_L_WORSTERR:	.BLKB	8
00018	AED_L_PAGewidth:	.BLKB	4
0001C	AED_L_PAGESIZE:	.BLKB	4
00020	AED_B_COLUMN:	.BLKB	4
00021		.BLKB	1
00024	AED_B_LINE:	.BLKB	3
00025		.BLKB	1
00028	AED_B_SAVE COL:	.BLKB	3
00029		.BLKB	1
0002C	AED_B_SAVE LIN:	.BLKB	3
0002D		.BLKB	1
00030	AED_Q_LINETABLE:	.BLKB	3
0003C	AED_L_CURACE:	.BLKB	12
00040	AED_L_FIRSTLINE:	.BLKB	4
00044	AED_L_LASTLINE:	.BLKB	4
00048	AED_L_BEGINLINE:	.BLKB	4
0004C	AED_W_INPUTLEN:	.BLKB	4
0004E		.BLKB	2
00050	AED_Q_DEL ACE:	.BLKB	2
00058	AED_Q_DEL LINE:	.BLKB	8
00060	AED_Q_DEL WORD:	.BLKB	8
00068	AED_B_DEL CHAR:	.BLKB	8
00069		.BLKB	1
0006C	AED_A_ACLBUFFER:	.BLKB	3
00070	AED_Q_OUTLINE:	.BLKB	4
00078	AED_W_OBJCHAN:	.BLKB	8
0007A		.BLKB	2
0007C	AED_W_TERMIN:	.BLKB	2
0007E		.BLKB	2
00080	AED_W_TERMOUT:	.BLKB	2
00082		.BLKB	2
00084	AED_W_IOSB:	.BLKB	2
0008C	AED_L_STATUS:	.BLKB	8


```
00090 AED_B_FIELD: .BLKB 4
00091 .BLKB 1
00094 AED_W_FIELDBEG: .BLKB 3
00096 .BLKB 2
00098 AED_W_FIELDEND: .BLKB 2
0009A .BLKB 2
0009C AED_B_ITEM: .BLKB 1
0009D .BLKB 3
000A0 AED_W_ITEMBEG: .BLKB 2
000A2 .BLKB 2
000A4 AED_W_ITEMEND: .BLKB 2
000A6 .BLKB 2
000A8 AED_B_ACETYPE: .BLKB 1
000A9 .BLKB 3
000AC AED_W_JOURNAL: .BLKB 2
000AE .BLKB 2
000B0 AED_T_CURLINE: .BLKB 532
002C4 AED_W_TOTALSIZE: .BLKB 2
002C6 .BLKB 2
002C8 JOURNAL_FAB: .BLKB 80
00318 JOURNAL_NAM: .BLKB 96
00378 JOURNAL_RAB: .BLKB 68
003BC JOURNAL_XABPRO: .BLKB 88
00414 JOURNAL_BUFFER: .BLKB 10
0041E .BLKB 2
00420 JOURNAL_INDEX: .BLKB 4
00424 RECOVER_FAB: .BLKB 80
00474 RECOVER_NAM: .BLKB 96
004D4 RECOVER_RAB: .BLKB 68
00518 RECOVER_BUFFER: .BLKB 10
00522 .BLKB 2
00524 RECOVER_INDEX: .BLKB 4
.PSECT $OWNS,NOEXE,2
```

00000 BUFFER_INDEX:
 .BKKB 4
00004 ECHO_DESC:
 .BKKB 8
0000C TEMP_LINE:
 .BKKB 4
00010 REMOVED_LINE:
 .BKKB 4
00014 REMOVED_ACE:
 .BKKB 4
00018 NEW_TEXT_LINE:
 .BKKB 4
0001C CHAR_PROCESSED:
 .BKKB 4
00020 APPEND_INDEX:
 .BKKB 4
00024 DUMMY_LINE:
 .BKKB 4
00028 TERM_CHAR:
 .BKKB 1
00029 .BKKB 3
0002C SEARCH_SIZE:
 .BKKB 2
0002E .BKKB 2
00030 SEARCH_STRING:
 .BKKB 512

SEGMENT_SIZE= AED_T_CURLINE+8
INPUT_BUFFER= AED_T_CURLINE+20
.EXTRN CLISGET_VALUE, CLISPRESENT
.EXTRN LIB\$FREE_VM, LIB\$GET_VM
.EXTRN LIB\$PARSE, SCR\$DOWN_SCROLL
.EXTRN SCR\$ERASE_LINE, SCR\$ERASE_PAGE
.EXTRN SCR\$SET_CURSOR, SCR\$SET_SCROLL
.EXTRN SCR\$UP_SCROLL, AED\$OBJLOCKED
.EXTRN AED\$BADKEEP, AED\$_LOCATERR
.EXTRN AED\$_INIREADERR
.EXTRN AED\$_JOUWRITER
.EXTRN AED\$_JOUOPENOUT
.EXTRN AED\$_JOUCLOSEOUT
.EXTRN AED\$_RECREADER
.EXTRN AED\$_RECOPENIN, AED\$_RECLOSEIN
.EXTRN AED\$_BADUIC, AED\$_BADGRPMEM
.EXTRN AED\$_SYNTAX, AED\$_BADTYPE
.EXTRN AED\$_NOITEMSEL, AED\$_MUSTENTER
.EXTRN AED\$_INIOPENIN, AED\$_INICLOSIN
.EXTRN AED\$_DEFSYNTAX, AED\$_NODELETE
.EXTRN AED\$_NOMODIFY, AED\$_NOHIDDEN
.EXTRN AED\$_DUPLICATE, AED\$_NOCOMBINE
.EXTRN AED\$_NODEFAULT, AED\$_NOCTRLCHAR
.EXTRN AED\$_NOTFOUND, AED\$_CONTROL_C
.EXTRN AED\$_ACLUPDATED
.EXTRN AED\$_NOCHANGE, AED_PUTOUTPUT
.EXTRN AED_GIVEHELP, AED_UPDATEACL
.EXTRN AED_SET_CURSOR, AED_SELECTFIELD
.EXTRN AED_SELECTITEM, AED_SETACETYPE
.EXTRN AED_COMPRESS, AED_POSITION

				OFFC 00000							
08		00		5B	00000000G	00	9E	00002			
				5A	0000G	CF	9E	00009			
				59	00000000G	00	9E	0000E			
				58	0000'	CF	9E	00015			
				57	0000'	CF	9E	0001A			
				5E		04	C2	0001F			
				6E		00	2C	00022			
					04	A8		00027			
	60	A7	04	A8		08	28	00029			
	58	A7	04	A8		08	28	0002F			
					68	A7	94	00035			
						68	D4	00038			
			50	A7	50	A7	9E	0003A			
			54	A7	50	A7	9E	0003F			
				67	1040	8F	A8	00044			
				50	30	A7	9E	00049			
				50	30	A7	D1	0004D			
						69	12	00051			
				67	4020	8F	A8	00053			
					00B8	C7	B4	00058			
					02C4	C7	B4	0005C			
			34	B7	00B0	C7	0E	00060			
				50	00B0	C7	9E	00066			
			44	A7		50	D0	0006B			
			40	A7		50	D0	0006F			
			0A	A0		01	B0	00073			
					3C	A7	D4	00077			
					01	A7	95	0007A			
						03	19	0007D			
					00AE	31		0007F			
					00A8	C7	94	00082	1\$:		
			02	A7		08	8A	00086			
						58	DD	0008A			
			0000G	CF		01	FB	0008C			
			04	A8	00B8	C7	B0	00091			
			08	A8	00C4	C7	9E	00097			
						01	DD	0009D			
				7E	24	A7	9A	0009F			
				69		02	FB	000A3			
					04	A8	9F	000A6			
				6A		01	FB	000A9			
				7E	00B8	C7	3C	000AC			
						6E	D6	000B1			
				7E	24	A7	9A	000B3			
				6B		02	FB	000B7			
						62	11	000BA			
					30	A7	DD	000BC	2\$:		
			0000G	CF		01	FB	000BF			
.EXTRN AED_COPSEGMENT, AED_REPSEGMENT											
.EXTRN AED_SEGSPLIT, AED_SEGCOMBINE											
.EXTRN AED_DECODEKEY, LIB\$SIGNAL											
.EXTRN SYS\$SETEXV											
.PSECT \$CODE\$, NOWRT, 2											
.ENTRY AED_PROCESSACL, Save R2,R3,R4,R5,R6,R7,R8,- R9,R10,R11 0648											
MOVAB SCR\$ERASE_LINE, R11											
MOVAB AED_PUTOUTPUT, R10											
MOVAB SCR\$SET_CURSOR, R9											
MOVAB BUFFER_INDEX, R8											
MOVAB AED_L_FLAGS, R7											
SUBL2 #4, -SP											
MOVCS #0, (SP), #0, #8, ECHO_DESC 0696											
MOVCS #8, ECHO_DESC, AED_Q_DEL_WORD 0697											
MOVCS #8, ECHO_DESC, AED_Q_DEL_LINE 0698											
CLRB AED_B_DEL_CHAR 0699											
CLRL BUFFER_INDEX 0700											
MOVAB AED_Q_DEL_ACE, AED_Q_DEL_ACE 0704											
MOVAB AED_Q_DEL_ACE, AED_Q_DEL_ACE+4 0705											
BISW2 #4180, AED_L_FLAGS 0707											
MOVAB AED_Q_LINETABLE, R0 0714											
CML AED_Q_LINETABLE, R0											
BNEQ 2\$											
BISW2 #16416, AED_L_FLAGS 0718											
CLRW SEGMENT_SIZE 0719											
CLRW AED_W_TOTALSIZE											
INSQUE AED_T_CURLINE, @AED_Q_LINETABLE+4 0720											
MOVAB AED_T_CURLINE, R0 0721											
MOVL R0, AED_L_LASTLINE											
MOVL R0, AED_L_FIRSTLINE											
MOVW #1, 10(R0) 0722											
CLRL AED_L_CURACE 0723											
TSTB AED_L_FLAGS+1 0724											
BLSS 1\$											
BRW 7\$											
CLRB AED_B_ACETYPE 0727											
BICB2 #8, -AED_L_FLAGS+2 0728											
PUSHL R8 0729											
CALLS #1, AED_SELECTFIELD											
MOVW AED_T_CURLINE+8, ECHO_DESC 0730											
MOVAB AED_T_CURLINE+20, ECHO_DESC+4 0731											
PUSHL #1 0732											
MOVZBL AED_B_LINE, -(SP)											
CALLS #2, -SCR\$SET_CURSOR											
PUSHAB ECHO_DESC 0733											
CALLS #1, AED_PUTOUTPUT											
MOVZWL SEGMENT_SIZE, -(SP) 0734											
INCL (SP)											
MOVZBL AED_B_LINE, -(SP)											
CALLS #2, -SCR\$ERASE_LINE											
BRB 6\$ 0735											
PUSHL AED_Q_LINETABLE 0741											
CALLS #1, -AED_COPSEGMENT											

	30	A7	00B0	C7	0E	000C4	INSQUE	AED_T_CURLINE, AED_Q_LINETABLE	: 0742
		50	00B0	C7	9E	000CA	MOVAB	AED-T-CURLINE, R0	0743
	44	A7		50	D0	000CF	MOVL	R0, AED-L-LASTLINE	
	40	A7		50	D0	000D3	MOVL	R0, AED-L-FIRSTLINE	
		51	40	A7	D0	000D7	MOVL	AED L FIRSTLINE, R1	0744
	02C4	C7	08	A1	B0	000DB	MOVW	8(RT), AED_W_TOTALSIZE	
		50	44	A7	D0	000E1	MOVL	AED_L_LASTLINE, R0	0745
1F	0A	A0		01	E0	000E5	BBS	#1, -10(R0), 5\$	
		52	00B0	C7	9E	000EA	MOVAB	AED_T_CURLINE, R2	0748
		52		50	D1	000EF	CMPL	R0, -R2	
				04	12	000F2	BNEQ	4\$	
	44	A7		60	D0	000F4	MOVL	(R0), AED L LASTLINE	0749
	44	A7	44	B7	D0	000FB	MOVL	AED L LASTLINE, AED_L_LASTLINE	0750
		50	44	A7	D0	000FD	MOVL	AED L LASTLINE, R0	0751
	02C4	C7	08	A0	A0	00101	ADDW2	8(R0), AED_W_TOTALSIZE	
				DC	11	00107	BRB	3\$	0745
	3C	A7	0C	A1	D0	00109	MOVL	12(R1), AED_L_CURACE	0753
			01	A7	95	0010E	TSTB	AED_L_FLAGS+1-	0754
				1D	18	00111	BGEQ	7\$	
	02	A7		08	88	00113	BISB2	#8, AED_L_FLAGS+2	0757
				58	DD	00117	PUSHL	R8	0758
	0000G	CF		01	FB	00119	CALLS	#1, AED_SELECTFIELD	
20	A7	68		01	81	0011E	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	0759
		7E	20	A7	9A	00123	MOVZBL	AED_B_COLUMN, -(SP)	0760
		7E	24	A7	9A	00127	MOVZBL	AED_B_LINE, -(SP)	
	0000G	CF		02	FB	0012B	CALLS	#2, AED_SET_CURSOR	
	48	A7	30	A7	D0	00130	MOVL	AED_Q_LINETABLE, AED_L_BEGINLINE	0763
	0000G	CF		00	FB	00135	CALLS	#0, AED_DECODEKEY	0771
	28	A8		50	90	0013A	MOVB	R0, TERM_CHAR	
				01	12	0013E	BNEQ	9\$	0772
					04	00140	RET		
	20	67		06	E1	00141	BBC	#6, AED_L_FLAGS, 10\$	0773
1C		67		03	E1	00145	BBC	#3, AED_L_FLAGS, 10\$	
				01	DD	00149	PUSHL	#1	0776
				15	DD	0014B	PUSHL	#21	
	00000000G	00		02	FB	0014D	CALLS	#2, SCRSEERASE PAGE	
		68		01	C1	00154	ADDL3	#1, BUFFER_INDEX, -(SP)	0777
7E		7E	24	A7	9A	00158	MOVZBL	AED B LINE, -(SP)	
	0000G	CF		02	FB	0015C	CALLS	#2, AED SET CURSOR	
		67	40	8F	8A	00161	BICB2	#64, AED L FLAGS	0778
03	02	A7		05	E0	00165	BBS	#5, AED_L_FLAGS+2, 11\$	0783
				01B2	31	0016A	BRW	68\$	
	28	01	28	A8	8F	0016D	CASEB	TERM CHAR, #1, #40	0786
				0117		00172	.WORD	49\$-12\$, -	
00F4	012A	0124		00FB		0017A		51\$-12\$, -	
0109	009B	007F		0102		00182		52\$-12\$, -	
00CC	00BD	0071		00C5		0018A		41\$-12\$, -	
006A	00BE	00D2		0086		00192		43\$-12\$, -	
00E6	00ED	00D8		0078		0019A		21\$-12\$, -	
0135	0110	013C		FFC3		001A2		25\$-12\$, -	
00A9	00A2	FFC3		00B0		001AA		46\$-12\$, -	
016E	0167	00B7		00DF		001B2		45\$-12\$, -	
0094	0063	005C		0143		001BA		19\$-12\$, -	
015F	0151	0147		0055		001C2		23\$-12\$, -	

					30\$-12\$,-		
					17\$-12\$,-		
					22\$-12\$,-		
					35\$-12\$,-		
					39\$-12\$,-		
					37\$-12\$,-		
					20\$-12\$,-		
					56\$-12\$,-		
					47\$-12\$,-		
					55\$-12\$,-		
					8\$-12\$,-		
					8\$-12\$,-		
					26\$-12\$,-		
					27\$-12\$,-		
					28\$-12\$,-		
					29\$-12\$,-		
					65\$-12\$,-		
					66\$-12\$,-		
					36\$-12\$,-		
					14\$-12\$,-		
					15\$-12\$,-		
					24\$-12\$,-		
					58\$-12\$,-		
					59\$-12\$,-		
					62\$-12\$,-		
					64\$-12\$,-		
					13\$-12\$,-		
		FF6E	31	001C4	BRW	8\$	
0000V	CF	00	FB	001C7 13\$:	CALLS	#0, ACT_RUB_CHR	1003
		97	11	001CC	BRB	10\$	0791
0000V	CF	00	FB	001CE 14\$:	CALLS	#0, ACT_RUB_WRD	0795
		67	11	001D3	BRB	32\$	
0000V	CF	00	FB	001D5 15\$:	CALLS	#0, ACT_RUB_BOL	0799
		89	11	001DA 16\$:	BRB	10\$	
0000V	CF	00	FB	001DC 17\$:	CALLS	#0, ACT_DEL_CHR	0801
		82	11	001E1 18\$:	BRB	10\$	
0000V	CF	00	FB	001E3 19\$:	CALLS	#0, ACT_DEL_WRD	0805
		7A	11	001E8	BRB	40\$	
0000V	CF	00	FB	001EA 20\$:	CALLS	#0, ACT_DEL_EOL	0811
		7A	11	001EF	BRB	42\$	
0000V	CF	00	FB	001F1 21\$:	CALLS	#0, ACT_DEL_ACE	0817
		7A	11	001F6	BRB	44\$	
0000V	CF	00	FB	001F8 22\$:	CALLS	#0, ACT_UNDEL_CHR	0823
		5E	11	001FD	BRB	38\$	
0000V	CF	00	FB	001FF 23\$:	CALLS	#0, ACT_UNDEL_WRD	0825
		57	11	00204	BRB	38\$	
0000V	CF	00	FB	00206 24\$:	CALLS	#0, ACT_UNDEL_LIN	0827
		50	11	0020B	BRB	38\$	
0000V	CF	00	FB	0020D 25\$:	CALLS	#0, ACT_UNDEL_ACE	0831
		73	11	00212	BRB	48\$	
0000V	CF	00	FB	00214 26\$:	CALLS	#0, ACT_UP	0839
		6C	11	00219	BRB	48\$	
0000V	CF	00	FB	0021B 27\$:	CALLS	#0, ACT_DOWN	0845
		65	11	00220	BRB	48\$	
0000V	CF	00	FB	00222 28\$:	CALLS	#0, ACT_RIGHT	0849
		7C	11	00227	BRB	54\$	
0000V	CF	00	FB	00229 29\$:	CALLS	#0, ACT_LEFT	0851

0000V	CF	75	11	0022E	BRB	54\$		
		00	FB	00230	CALLS	#0,	ACT_TOP	0855
		7C	11	00235	BRB	57\$		
0000V	CF	00	FB	00237	CALLS	#0,	ACT_BOTTOM	0861
		75	11	0023C	BRB	57\$		
01	A7	01	8A	0023E	BICB2	#1,	AED_L_FLAGS+1	0869
		49	11	00242	BRB	50\$		0870
01	A7	01	88	00244	BISB2	#1,	AED_L_FLAGS+1	0876
		43	11	00248	BRB	50\$		0877
0000V	CF	00	FB	0024A	CALLS	#0,	ACT_MOVE_WRD	0883
		6F	11	0024F	BRB	61\$		
0000V	CF	00	FB	00251	CALLS	#0,	ACT_MOVE_BOL	0885
		82	11	00256	BRB	16\$		
0000V	CF	00	FB	00258	CALLS	#0,	ACT_MOVE_EOL	0887
		82	11	0025D	BRB	18\$		
0000V	CF	00	FB	0025F	CALLS	#0,	ACT_MOVE_ACE	0891
		64	11	00264	BRB	63\$		
0000V	CF	00	FB	00266	CALLS	#0,	ACT_FIND_STR	0897
		5D	11	00268	BRB	63\$		
0000V	CF	00	FB	0026D	CALLS	#0,	ACT_FIND_NXT	0903
		56	11	00272	BRB	63\$		
0000V	CF	00	FB	00274	CALLS	#0,	ACT_ADV_FIELD	0911
		4F	11	00279	BRB	63\$		
0000V	CF	00	FB	0027B	CALLS	#0,	ACT_SEL_FIELD	0917
		48	11	00280	BRB	63\$		
0000V	CF	00	FB	00282	CALLS	#0,	ACT_SEL_ITEM	0923
		41	11	00287	BRB	63\$		
01	A7	08	88	00289	BISB2	#8,	AED_L_FLAGS+1	0931
02	A7	20	8A	0028D	BICB2	#32,	AED_L_FLAGS+2	0932
		28	AB	94	CLRB	TERM_CHAR		0933
		2A	11	00294	BRB	61\$		0786
02	A7	10	8A	00296	BICB2	#16,	AED_L_FLAGS+2	0938
		04	11	0029A	BRB	53\$		0939
02	A7	10	88	0029C	BISB2	#16,	AED_L_FLAGS+2	0944
0000V	CF	00	FB	002A0	CALLS	#0,	ACT_RECV	0945
		19	11	002A5	BRB	61\$		0786
0000V	CF	00	FB	002A7	CALLS	#0,	ACT_ENTER	0950
		1C	11	002AC	BRB	63\$		
0000V	CF	00	FB	002AE	CALLS	#0,	ACT_INSERT	0956
		15	11	002B3	BRB	63\$		
		7E	D4	002B5	CLRL	-(SP)		0960
		02	11	002B7	BRB	60\$		
0000V	CF	01	DD	002B9	PUSHL	#1		0962
		01	FB	002BB	CALLS	#1,	ACT_REFRESH	
		FEA2	31	002C0	BRW	10\$		
		7E	D4	002C3	CLRL	-(SP)		0966
0000V	CF	01	FB	002C5	CALLS	#1,	ACT_EXIT	
	56	50	D0	002CA	MOVL	R0,	LOCAL STATUS	
	F0	56	E8	002CD	BLBS	LOCAL STATUS, 61\$		0967
		04	002D0	RET				
0000V	CF	01	DD	002D1	PUSHL	#1		0972
		01	FB	002D3	CALLS	#1,	ACT_EXIT	
		04	002D8	RET				0971
02	A7	80	8F	8C	XORB2	#128,	AED_L_FLAGS+2	0978
		3D	11	002DE	BRB	67\$		0979
		5E	DD	002E0	PUSHL	SP		0989
		7E	7C	002E2	CLRQ	-(SP)		

00000000G	00	7E	D4	002E4	CLRL	-(SP)	
		04	FB	002E6	CALLS	#4, SYS\$SETEXV	
		6E	D5	002ED	TSTL	PREV_HANDLER	0990
		74	13	002EF	BEQL	71\$	
	08	7E	7C	002F1	CLRL	-(SP)	0991
		AE	DD	002F3	PUSHL	PREV_HANDLER	
00000000G	00	7E	D4	002F6	CLRL	-(SP)	
		04	FB	002F8	CALLS	#4, SYS\$SETEXV	
		01	DD	002FF	PUSHL	#1	0995
	69	15	DD	00301	PUSHL	#21	
00000000G	00	02	FB	00303	CALLS	#2, SCR\$SET_CURSOR	
	7E	8F	3C	00306	MOVZWL	#132, -(SP)	0996
	00	01	FB	00308	CALLS	#1, LIB\$SIGNAL	
	7E	A7	9A	00312	MOVZBL	AED_B_COLUMN, -(SP)	0997
	7E	A7	9A	00316	MOVZBL	AED_B_LINE, -(SP)	
	69	02	FB	0031A	CALLS	#2, SCR\$SET_CURSOR	
		5C	11	0031D	BRB	72\$	0998
	28	A8	95	0031F	TSTB	TERM_CHAR	1010
		57	13	00322	BEQL	72\$	
	40	A7	D0	00324	MOVL	AED_L_FIRSTLINE, R0	1014
51	0A	04	E1	00328	BBC	#4, -10(R0), 73\$	
12	67	03	E1	0032D	BBC	#3, AED_L_FLAGS, 69\$	1017
		01	DD	00331	PUSHL	#1	
00000000G	00	15	DD	00333	PUSHL	#21	
		02	FB	00335	CALLS	#2, SCR\$ERASE_PAGE	
		01	DD	0033C	PUSHL	#1	
	69	15	DD	0033E	PUSHL	#21	
00000000G	00	02	FB	00340	CALLS	#2, SCR\$SET_CURSOR	
		8F	DD	00343	PUSHL	#AED\$NOMODIFY	
08	00	01	FB	00349	CALLS	#1, LIB\$SIGNAL	
	67	03	E1	00350	BBC	#3, AED_L_FLAGS, 70\$	
	7E	A7	9A	00354	MOVZBL	AED_B_COLUMN, -(SP)	
	7E	A7	9A	00358	MOVZBL	AED_B_LINE, -(SP)	
	69	02	FB	0035C	CALLS	#2, SCR\$SET_CURSOR	
		8F	D5	0035F	TSTL	#<AED\$NOMODIFY&7>	70\$:
00000000*	8F	14	13	00365	BEQL	72\$	71\$:
		00	ED	00367	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$NOMODIFY&7>	
		08	18	00371	BGEQ	72\$	
	14	A7	D0	00373	MOVL	#AED\$NOMODIFY, AED_L_WORSTERR	
		8F	D0	0037B	BRW	8\$	1018
		05	E0	0037E	BBS	#5, AED_L_FLAGS+2, 72\$	1023
F8	02	A7	9A	00383	MOVZBL	TERM_CHAR, R1	1024
		51	91	00387	CMPB	R1, #13	1026
	0D	03	13	0038A	BEQL	74\$	
		00DC	31	0038C	BRW	77\$	
		10	8A	0038F	BICB2	#16, AED_L_FLAGS+1	1031
		57	18	00393	BGEQ	75\$	1032
68	0088	C7	00	ED	CMPZV	#0, #16, SEGMENT_SIZE, BUFFER_INDEX	1033
		10	4E	14	BGTR	75\$	
		50	C7	9E	MOVAB	INPUT_BUFFER, R0	1036
		50	C0	003A3	ADDL2	BUFFER_INDEX, R0	
		2C	A0	91	CMPB	-1(R0), #44	
		40	13	003AA	BEQL	75\$	
		3D	A0	91	CMPB	-1(R0), #61	1037
			3A	13	BEQL	75\$	
		29	A0	91	CMPB	-1(R0), #41	1038
			34	13	BEQL	75\$	

30	02	A7	F8	003B8	BLBS	AED_L_FLAGS+2, 758	1039	
02	0090	C7	91	003BC	CMPB	AED_B_FIELD, #2	1040	
		29	1E	003C1	BGEQU	758		
	00B8	C7	B5	003C3	TSTW	SEGMENT_SIZE	1041	
		23	13	003C7	BEQL	758		
50	00C4	C7	9E	003C9	MOVAB	INPUT_BUFFER, R0	1043	
00 B840		2C	90	003CE	MOVW	#44, @BUFFER_INDEX[R0]	1044	
04 A8		01	B0	003D3	MOVW	#1, ECHO_DESC	1045	
08 A8	00 B840	9E	9E	003D7	MOVAB	@BUFFER_INDEX[R0], ECHO_DESC+4	1046	
	04 A8	9F	9F	003DD	PUSHAB	ECHO_DESC	1047	
6A		01	FB	003E0	CALLS	#1, AED_PUTOUTPUT		
	20	A7	96	003E3	INCB	AED_B_COLUMN	1048	
		68	D6	003E6	INCL	BUFFER_INDEX	1049	
	00B8	C7	B6	003E8	INCW	SEGMENT_SIZE	1050	
		7E	7C	003EC	CLRW	-(SP)	1056	
		01	DD	003EE	PUSHL	#1		
		58	DD	003F0	PUSHL	R8		
0000G CF		04	FB	003F2	CALLS	#4, AED_SEGSPLIT		
18 A8		50	D0	003F7	MOVL	R0, NEW_TEXT_LINE		
	01	A7	95	003FB	TSTB	AED_L_FLAGS+T	1060	
		56	1B	003FE	BGEQ	768		
4D 02	52	02	A7	E8	00400	BLBS	AED_L_FLAGS+2, 768	1061
	A7		03	E0	00404	BBS	#3, AED_L_FLAGS+2, 768	1062
	50	00B0	C7	9E	00409	MOVAB	AED_T_CORLINE, R0	1063
	50	44	A7	D1	0040E	CMPL	AED_L_LASTLINE, R0	
			42	12	00412	BNEQ	768	
	50	40	A7	D0	00414	MOVL	AED_L_FIRSTLINE, R0	1064
		0C	A0	D5	00418	TSTL	12(R0)	
			39	12	0041B	BNEQ	768	
	02 A7		08	8A	0041D	BICB2	#8, AED_L_FLAGS+2	1067
			58	DD	00421	PUSHL	R8	1068
0000G CF			01	FB	00423	CALLS	#1, AED_SELECTFIELD	
04 A8	00B8	C7	B0	00428	MOVW	AED_T_CORLINE+8, ECHO_DESC	1069	
08 A8	00C4	C7	9E	0042E	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	1070	
		01	DD	00434	PUSHL	#1	1071	
	7E	24	A7	9A	00436	MOVZBL	AED_B_LINE, -(SP)	
69			02	FB	0043A	CALLS	#2, SCRSET_CURSOR	
	04	A8	9F	0043D	PUSHAB	ECHO_DESC	1072	
6A		01	FB	00440	CALLS	#1, AED_PUTOUTPUT		
7E	00B8	C7	3C	00443	MOVZWL	SEGMENT_SIZE, -(SP)	1073	
		6E	D6	00448	INCL	(SP)		
	24	A7	9A	0044A	MOVZBL	AED_B_LINE, -(SP)		
		02	FB	0044E	CALLS	#2, SCRERASE_LINE		
20 A7	68	01	81	00451	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	1074	
	7E	20	A7	9A	00456	MOVZBL	AED_B_COLUMN, -(SP)	1076
	7E	24	A7	9A	0045A	MOVZBL	AED_B_LINE, -(SP)	
0000G CF		02	FB	0045E	CALLS	#2, AED_SET_CURSOR		
01 A7		40	8F	88	00463	BISB2	#64, AED_L_FLAGS+1	1077
			00E0	31	00468	BRW	888	1078
	20		51	91	0046B	CMPB	R1, #32	1089
			03	1E	0046E	BGEQU	788	
			FCC2	31	00470	BRW	88	
	67	80	8F	88	00473	BISB2	#128, AED_L_FLAGS	1090
01 A7			10	8A	00477	BICB2	#16, AED_C_FLAGS+1	1091
61 8F			51	91	0047B	CMPB	R1, #97	1092
			06	1F	0047F	BLSSU	798	
7A 8F			51	91	00481	CMPB	R1, #122	

					0C 1B 00485		BLEQU	808		
	E0	8F			51 91 00487	798:	CMPB	R1, #224		1093
					0A 1F 00488		BLSSU	818		
	FE	8F			51 91 0048D		CMPB	R1, #254		
					04 1A 00491		BGTRU	818		
	28	A8			20 82 00493	808:	SUBB2	#32, TERM_CHAR		1094
	18	A7			68 D1 00497	818:	CMPL	BUFFER_INDEX, AED_L_PAGEWIDTH		1099
					0F 18 00498		BGEQ	828		
				02	A7 95 0049D		TSTB	AED_L_FLAGS+2		1100
					15 19 004A0		BLSS	838		
18	A7	00B8	C7	10	00 ED 004A2		CMPZV	#0, #16, SEGMENT_SIZE, AED_L_PAGEWIDTH		1101
					0B 19 004AA		BLSS	838		
					7E 7C 004AC	828:	CLRQ	-(SP)		1102
					7E D4 004AE		CLRL	-(SP)		
					58 DD 004B0		PUSHL	R8		
	0000G	CF			04 FB 004B2		CALLS	#4, AED_SEGSPLIT		
	04	A8			01 B0 004B7	838:	MOVW	#1, ECHO_DESC		1104
	08	A8			28 A8 9E 004BB		MOVAB	TERM_CHAR, ECHO_DESC+4		1105
				04	A8 9F 004C0		PUSHAB	ECHO_DESC		1106
		6A			01 FB 004C3		CALLS	#1, AED_PUTOUTPUT		
		50			68 D0 004C6		MOVL	BUFFER_INDEX, R0		1111
50	00B8	C7		10	00 ED 004C9		CMPZV	#0, #16, SEGMENT_SIZE, R0		
					3E 15 004D0		BLEQ	848		
				02	A7 95 004D2		TSTB	AED_L_FLAGS+2		1112
					39 19 004D5		BLSS	848		
	04	A8	00B8	C7	50 A3 004D7		SUBW3	R0, SEGMENT_SIZE, ECHO_DESC		1115
		08	A8		00C4 C740 9E 004DE		MOVAB	INPUT_BUFFER[R0], ECHO_DESC+4		1116
					04 A8 9F 004E5		PUSHAB	ECHO_DESC		1117
		6A			01 FB 004E8		CALLS	#1, AED_PUTOUTPUT		
	7E	68			02 C1 004EB		ADDL3	#2, BUFFER_INDEX, -(SP)		1118
		7E			24 A7 9A 004EF		MOVZBL	AED_B_LINE, -(SP)		
		0000G	CF		02 FB 004F3		CALLS	#2, AED_SET_CURSOR		
		50			68 D0 004F8		MOVL	BUFFER_INDEX, R0		1119
	51	000001FF	8F		50 C3 004FB		SUBL3	R0, #51, R1		1121
51	00	00C4	C740		04 A8 2C 00503		MOVCS	ECHO_DESC, INPUT_BUFFER[R0], #0, R1, -		
					00C5 C740 0050C			INPUT_BUFFER+1[R0]		
		51			28 A8 9A 00510	848:	MOVZBL	TERM_CHAR, R1		1124
	5B	8F			51 91 00514		CMPB	R1, #91		
					04 12 00518		BNEQ	858		
	02	A7			01 88 0051A		BISB2	#1, AED_L_FLAGS+2		
	5D	8F			51 91 0051E	858:	CMPB	R1, #93		1125
					04 12 00522		BNEQ	868		
	02	A7			01 8A 00524		BICB2	#1, AED_L_FLAGS+2		
	50				00C4 C7 9E 00528	868:	MOVAB	INPUT_BUFFER, R0		
	00	B840			51 90 0052D		MOVW	R1, #BUFFER_INDEX[R0]		1129
					68 D6 00532		INCL	BUFFER_INDEX		1130
	20	A7		68	01 81 00534		ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN		1131
					02 A7 95 00539		TSTB	AED_L_FLAGS+2		1135
					09 18 0053C		BGEQ	878		
68	00B8	C7		10	00 ED 0053E		CMPZV	#0, #16, SEGMENT_SIZE, BUFFER_INDEX		1136
					04 14 00545		BGTR	888		
		01	A7		00B8 C7 B6 00547	878:	INCW	SEGMENT_SIZE		1137
					08 8A 0054B	888:	BICB2	#8, AED_L_FLAGS+1		1138
					FBE3 31 0054F		BRW	88		0768
					04 00552		RET			1144

; Routine Size: 1363 bytes. Routine Base: \$CODE\$ + 0000

AEDSMAN
V04-000

AED_PROCESSACL - main processing loop

J 11
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 24
(3)

ACT_RUB_CHR - rubout a single character

```
696 1145 1 ZSBTTL 'ACT RUB CHR - rubout a single character'
697 1146 1 ROUTINE ACT_RUB_CHR =
698 1147 1
699 1148 1 **
700 1149 1
701 1150 1 FUNCTIONAL DESCRIPTION:
702 1151 1
703 1152 1 This routine deletes the character immediately preceeding the current
704 1153 1 cursor position. The deleted character is placed in storage for
705 1154 1 later retrieval.
706 1155 1
707 1156 1 CALLING SEQUENCE:
708 1157 1 ACT_RUB_CHR ()
709 1158 1
710 1159 1 INPUT PARAMETERS:
711 1160 1 none
712 1161 1
713 1162 1 IMPLICIT INPUTS:
714 1163 1 OWN storage
715 1164 1
716 1165 1 OUTPUT PARAMETERS:
717 1166 1 none
718 1167 1
719 1168 1 IMPLICIT OUTPUTS:
720 1169 1 none
721 1170 1
722 1171 1 ROUTINE VALUE:
723 1172 1 1 if successful
724 1173 1 error status otherwise
725 1174 1
726 1175 1 SIDE EFFECTS:
727 1176 1 The line segment table is updated as necessary, ACE line pointers
728 1177 1 are updated, and the object's ACL is updated as necessary.
729 1178 1
730 1179 1 --
731 1180 1
732 1181 2 BEGIN
733 1182 2
734 1183 2 LOCAL
735 1184 2 PREV_LINE : REF $BLOCK, ! Addr of previous segment
736 1185 2 COMBINED_LINE : REF $BLOCK; ! Addr of combined segments
737 1186 2
738 1187 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
739 1188 2
740 1189 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
741 1190 2 THEN
742 1191 2 BEGIN
743 1192 2 SIGNAL (AED$ NOMODIFY);
744 1193 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
745 1194 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
746 1195 2 TERM_CHAR = 0;
747 1196 2 RETURN 1;
748 1197 2 END;
749 1198 2 AED_L_FLAGS[AED_V_MODIFIED] = 1;
750 1199 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
751 1200 2
752 1201 2 IF .BUFFER_INDEX GTR 0
```

```
753 1202 2 THEN
754 1203 BEGIN
755 1204
756 1205 ! Delete the previous character.
757 1206
758 1207 AED_B_DEL_CHAR = .INPUT_BUFFER[.BUFFER_INDEX - 1];
759 1208 IF .AED_B_DEL_CHAR EQL '[' THEN AED_L_FLAGS[AED_V_OPENUIIC] = 0;
760 1209 IF .AED_B_DEL_CHAR EQL ']' THEN AED_L_FLAGS[AED_V_OPENUIIC] = 0;
761 1210 SCR$SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN - 1);
762 1211 IF .BUFFER_INDEX LSS .SEGMENT_SIZE
763 1212 THEN
764 1213 BEGIN
765 1214 ECHO_DESC[DESC$W_LENGTH] = .SEGMENT_SIZE - .BUFFER_INDEX;
766 1215 ECHO_DESC[DESC$A_POINTER] = BUFFER_CHAR;
767 1216 AED_PUTOUTPUT (ECHO_DESC);
768 1217 CH$COPY (.ECHO_DESC[DESC$W_LENGTH], BUFFER_CHAR,
769 1218 0
770 1219 512 - .BUFFER_INDEX + 1, INPUT_BUFFER[.BUFFER_INDEX - 1]);
771 1220 END;
772 1221 BUFFER_INDEX = .BUFFER_INDEX - 1;
773 1222 AED_B_COLUMN = .BUFFER_INDEX + 1;
774 1223 SEGMENT_SIZE = .SEGMENT_SIZE - 1;
775 1224 SCR$ERASE_LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
776 1225 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
777 1226 END
778 1227
779 1228 ! Combine the current and previous line segments.
780 1229
781 1230 ELSE AED_SEGCOMBINE (BUFFER_INDEX, 0);
782 1231
783 1232 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
784 1233 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
785 1234 TERM_CHAR = 0;
786 1235 RETURN 1;
787 1236
788 1237 1 END; ! End of routine ACT_RUB_CHR
```

03FC 00000 ACT_RUB_CHR:						
	59	00000000G	8F	D0	00002	Save R2,R3,R4,R5,R6,R7,R8,R9
	58	00000000G	00	9E	00009	MOVAB #AED\$ NOMODIFY, R9
	57	0000'	CF	9E	00010	SCR\$SET_CURSOR, R8
	56	0000'	CF	9E	00015	MOVAB BUFFER_INDEX, R7
	50	40	A6	D0	0001A	MOVAB AED_L_FLAGS, R6
49	0A	A0	04	E1	0001E	MOVAB AED_L_FIRSTLINE, R0
12		66	03	E1	00023	BBC #4, 10(R0), 4\$
			01	DD	00027	BBC #3, AED_L_FLAGS, 1\$
			15	DD	00029	PUSHL #1
			02	FB	0002B	PUSHL #21
	00000000G	00	01	DD	00032	CALLS #2, SCR\$ERASE_PAGE
			15	DD	00034	PUSHL #1
		68	02	FB	00036	PUSHL #21
			59	DD	00039	CALLS #2, SCR\$SET_CURSOR
					1\$:	PUSHL R9

1146
1189
1192

000000000*	8F	14	A6	00	01	FB	0003B	CALLS	#1, LIB\$SIGNAL	
				66	03	E1	00042	BBC	#3, AED_L_FLAGS, 2\$	
				7E	A6	9A	00046	MOVZBL	AED_B_COLUMN, -(SP)	
				7E	A6	9A	0004A	MOVZBL	AED_B_LINE, -(SP)	
				68	02	FB	0004E	CALLS	#2, SCR\$SET_CURSOR	
				000000000*	8F	D5	00051	TSTL	#<AED\$_NOMODIFY&7>	
					10	13	00057	BEQL	3\$	
				03	00	ED	00059	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOMODIFY&7>	
				14	A6	59	00063	BGEQ	3\$	
					00A9	31	00065	MOVL	R9, AED_L_WORSTERR	
				66	80	8F	00069	BRW	10\$	1193
				01	A6	10	0006C	BISB2	#128, AED_L_FLAGS	1198
				50	67	D0	00070	BICB2	#16, AED_L_FLAGS+1	1199
					03	14	00074	MOVL	BUFFER_INDEX, R0	1201
					0090	31	00077	BGTR	5\$	
				68	A6	00C3	00079	BRW	9\$	
				58	8F	68	0007C	MOVB	INPUT_BUFFER-1[R0], AED_B_DEL_CHAR	1207
					04	A6	00083	CMPB	AED_B_DEL_CHAR, #91	1208
				02	A6	01	00088	BNEQ	6\$	
				50	8F	68	0008A	BICB2	#1, AED_L_FLAGS+2	
					04	A6	0008E	CMPB	AED_B_DEL_CHAR, #93	1209
				02	A6	01	00093	BNEQ	7\$	
					7E	20	00095	BICB2	#1, AED_L_FLAGS+2	
					6E	A6	00099	MOVZBL	AED_B_COLUMN, -(SP)	1210
					7E	24	0009D	DECL	(SP)	
				50	02	FB	0009F	MOVZBL	AED_B_LINE, -(SP)	
					67	D0	000A3	CALLS	#2, SCR\$SET_CURSOR	
					00	D0	000A6	MOVL	BUFFER_INDEX, R0	1211
					2E	15	000A9	CMPZV	#0, #18, SEGMENT_SIZE, R0	
					50	A3	000B0	BLEQ	8\$	
					00C4	9E	000B2	SUBW3	R0, SEGMENT_SIZE, ECHO_DESC	1214
					04	9F	000B9	MOVAB	INPUT_BUFFER[R0], ECHO_DESC+4	1215
					01	FB	000C0	PUSHAB	ECHO_DESC	1216
					67	D0	000C3	CALLS	#1, AED_PUTOUTPUT	
					50	C3	000C8	MOVL	BUFFER_INDEX, R0	1217
					04	A7	000CB	SUBL3	R0, #5T3, R1	1219
					00C3	2C	000D3	MOVC5	ECHO_DESC, INPUT_BUFFER[R0], #0, R1, -	
					67	D7	000DC		INPUT_BUFFER-1[R0]	
					01	81	000E0	DECL	BUFFER_INDEX	1221
					00B8	C6	000E2	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	1222
					00B8	C6	000E7	DECW	SEGMENT_SIZE	1223
					6E	D6	000EB	MOVZWL	SEGMENT_SIZE, -(SP)	1224
					7E	A6	000F0	INCL	(SP)	
					00000000G	02	000F2	MOVZBL	AED_B_LINE, -(SP)	
					7E	02	000F6	CALLS	#2, SCR\$ERASE_LINE	
					7E	A6	000FD	MOVZBL	AED_B_COLUMN, -(SP)	1225
					0000G	A6	00101	MOVZBL	AED_B_LINE, -(SP)	
						02	00105	CALLS	#2, AED_SET_CURSOR	
						09	0010A	BRB	10\$	1201
						7E	0010C	CLRL	-(SP)	1230
						57	0010E	PUSHL	R7	
						02	00110	CALLS	#2, AED_SEGCOMBINE	
						8F	00115	BICW2	#8200, AED_L_FLAGS+1	1233
						A7	0011B	CLRB	TERM_CHAR	1234
						01	0011E	MOVL	#1, R0	1235
						04	00121	RET		1237

AED\$MAIN
V04-000

ACT_RUB_CHR - rubout a single character

N 11
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 28
(4)

; Routine Size: 290 bytes, Routine Base: \$CODE\$ + 0553

ACT_RUB_WRD - rubout previous word

```
790 1238 1 XSBTTL 'ACT RUB WRD - rubout previous word'
791 1239 1 ROUTINE ACT_RUB_WRD =
792 1240 1
793 1241 1 ++
794 1242 1
795 1243 1 FUNCTIONAL DESCRIPTION:
796 1244 1
797 1245 1 This routine deletes the word (all characters stopping with the first
798 1246 1 non alphanumeric character) immediately preceeding the current cursor
799 1247 1 position. The deleted word is placed in storage for later retrieval.
800 1248 1
801 1249 1 CALLING SEQUENCE:
802 1250 1 ACT_RUB_WRD ()
803 1251 1
804 1252 1 INPUT PARAMETERS:
805 1253 1 none
806 1254 1
807 1255 1 IMPLICIT INPUTS:
808 1256 1 DWN storage
809 1257 1
810 1258 1 OUTPUT PARAMETERS:
811 1259 1 none
812 1260 1
813 1261 1 IMPLICIT OUTPUTS:
814 1262 1 none
815 1263 1
816 1264 1 ROUTINE VALUE:
817 1265 1 1 if successful
818 1266 1 error status otherwise
819 1267 1
820 1268 1 SIDE EFFECTS:
821 1269 1 The line segment table is updated as necessary, ACE line pointers
822 1270 1 are updated, and the object's ACL is updated as necessary.
823 1271 1
824 1272 1 --
825 1273 1
826 1274 2 BEGIN
827 1275 2
828 1276 2 LOCAL
829 1277 2 DEL_WORD_BEGIN, ! Beginning offset of word
830 1278 2 DEL_WORD_END; ! End offset of word
831 1279 2
832 1280 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
833 1281 2
834 1282 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
835 1283 2 THEN
836 1284 2 BEGIN
837 1285 2 SIGNAL (AED$ NOMODIFY);
838 1286 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
839 1287 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
840 1288 2 TERM_CHAR = 0;
841 1289 2 RETURN 1;
842 1290 2 END;
843 1291 2
844 1292 2 ! Deallocate anything in the saved word buffer.
845 1293 2
846 1294 2 IF .AED_Q_DEL_WORDS[DSCSW_LENGTH] NEQ 0
```

```

847 P 1295 2 THEN DEALLOCATE (.AED_Q_DEL_WORD[DSC$W_LENGTH],
848 1296 2 AED_Q_DEL_WORD[DSC$A_POINTER]);
849 1297 2 AED_Q_DEL_WORD[DSC$W_LENGTH] = 0;
850 1298 2
851 1299 2 AED_L_FLAGS[AED_V_MODIFIED] = 1; ! ACE has been modified
852 1300 2 AED_L_FLAGS[AED_V_FIRST_CHAR] = 0;
853 1301 2
854 1302 2 IF .BUFFER_INDEX GTR 0
855 1303 2 THEN
856 1304 2 BEGIN
857 1305 2
858 1306 2 ! Delete the previous word.
859 1307 2
860 1308 2 DEL_WORD_END = .BUFFER_INDEX;
861 1309 2 BUFFER_INDEX = .BUFFER_INDEX - 2; ! Backup over delimiter
862 1310 2 IF .BUFFER_INDEX GEQ 0
863 1311 2 THEN
864 1312 2 BEGIN
865 1313 2 WHILE (.BUFFER_CHAR GEQ 'A' AND .BUFFER_CHAR LEQ 'Z')
866 1314 2 OR (.BUFFER_CHAR GEQ '0' AND .BUFFER_CHAR LEQ '9')
867 1315 2 DO
868 1316 2 BEGIN
869 1317 2 BUFFER_INDEX = .BUFFER_INDEX - 1;
870 1318 2 IF .BUFFER_INDEX LSS 0 THEN EXITLOOP;
871 1319 2 END;
872 1320 2 END
873 1321 2 ELSE BUFFER_INDEX = -1;
874 1322 2 BUFFER_INDEX = .BUFFER_INDEX + 1; ! First char of word
875 1323 2 DEL_WORD_BEGIN = .BUFFER_INDEX;
876 1324 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
877 1325 2 AED_Q_DEL_WORD[DSC$W_LENGTH] = .DEL_WORD_END - .DEL_WORD_BEGIN;
878 P 1326 2 AED_L_STATUS = ALLOCATE (.AED_Q_DEL_WORD[DSC$W_LENGTH],
879 1327 2 AED_Q_DEL_WORD[DSC$A_POINTER]);
880 1328 2
881 1329 2 IF NOT .AED_L_STATUS
882 1330 2 THEN
883 1331 2 BEGIN
884 1332 2 SIGNAL (.AED_L_STATUS);
885 1333 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
886 1334 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
887 1335 2 TERM_CHAR = 0;
888 1336 2 RETURN 0;
889 1337 2 END;
890 1338 2 CH$MOVE (.AED_Q_DEL_WORD[DSC$W_LENGTH],
891 1339 2 INPUT_BUFFER[.DEL_WORD_BEGIN],
892 1340 2 .AED_Q_DEL_WORD[DSC$A_POINTER]);
893 1341 2 AED_L_FLAGS[AED_V_ROBWORD] = 1;
894 1342 2 ECHO_DESC[DSC$W_LENGTH] = .SEGMENT_SIZE - .DEL_WORD_END;
895 1343 2 ECHO_DESC[DSC$A_POINTER] = INPUT_BUFFER[.DEL_WORD_END];
896 1344 2 IF .ECHO_DESC[DSC$W_LENGTH] GEQ 1
897 1345 2 THEN
898 1346 2 BEGIN
899 1347 2 SCR$SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
900 1348 2 AED_PUTOUTPUT (ECHO_DESC);
901 1349 2 END;
902 1350 2 CH$COPY (.ECHO_DESC[DSC$W_LENGTH], INPUT_BUFFER[.DEL_WORD_END],
903 1351 2 0
903 1351 2 512 - .BUFFER_INDEX, INPUT_BUFFER[.DEL_WORD_BEGIN]);
```



```
! End of routine ACT_RUB_WRD
```

Offset	Hex	Symbol	Disassembly	Comment	Address
00000000	5B 00000000G	00	9E 00002	WORD	1239
00000001	5A 0000	CF	9E 00009	MOVAB	
00000002	59 0000	CF	9E 0000E	MOVAB	
00000003	5E	04	C2 00013	SUBL2	
00000004	50 40	A9	D0 00016	MOVL	1282
00000005	A0	04	E1 0001A	BBC	
00000006	69	03	E1 0001F	BBC	1285
00000007		01	DD 00023	PUSHL	
00000008		15	DD 00025	PUSHL	
00000009	00	02	FB 00027	CALLS	
0000000A		01	DD 0002E	PUSHL	
0000000B		15	DD 00030	PUSHL	
0000000C	68	02	FB 00032	CALLS	
0000000D	00000000G	8F	DD 00035	1\$: PUSHL	
0000000E	00	01	FB 0003B	CALLS	
0000000F	69	03	E1 00042	BBC	
00000010	7E 20	A9	9A 00046	MOVZBL	
00000011	7E 24	A9	9A 0004A	MOVZBL	
00000012	68	02	FB 0004E	CALLS	
00000013	00000000*	8F	D5 00051	2\$: TSTL	
00000014		14	13 00057	BEQL	
00000015	03	00	ED 00059	CMPZV	
00000016		08	18 00063	BGEQ	
00000017	14	A9	00000000G 8F D0 00065	MOVL	
00000018		01	65 31 0006D	3\$: BRW	1286
00000019	50 60	A9	3C 00070	4\$: MOVZWL	1294
0000001A		11	13 00074	BEQL	
0000001B	64	A9	9F 00076	PUSHAB	1296
0000001C	04	AE	50 D0 00079	MOVL	
0000001D		04	AE 9F 0007D	PUSHAB	
0000001E	00000000G	00	02 FB 00080	CALLS	
0000001F		60	A9 B4 00087	5\$: CLRW	1297
00000020	69 80	8F	88 0008A	BISB2	1299
00000021	01	A9	10 8A 0008E	BICB2	1300
00000022	50	6A	D0 00092	MOVL	1302

				03	14	00095	BGTR	6\$		
				0132	31	00097	BRW	18\$		
		57		50	D0	0009A	6\$:	RO, DEL_WORD_END		1308
		6A		02	C2	0009D	SUBL2	#2, BUFFER_INDEX		1309
				28	19	000A0	BLSS	10\$		1310
		50		6A	D0	000A2	MOVL	BUFFER_INDEX, RO		1313
		51		00C4	C940	9A 000A5	7\$:	MOVZBL INPUT_BUFFER[RO], R1		
	41	8F		51	91	000AB	CMPB	R1, #85		
				06	1F	000AF	BLSSU	8\$		
	5A	8F		51	91	000B1	CMPB	R1, #90		
				0A	1B	000B5	BLEQU	9\$		
		30		51	91	000B7	8\$:	CMPB R1, #48		1314
				11	1F	000BA	BLSSU	11\$		
		39		51	91	000BC	CMPB	R1, #57		
				0C	1A	000BF	BGTRU	11\$		
		50		6A	D7	000C1	9\$:	DECL BUFFER_INDEX		1317
				6A	D0	000C3	MOVL	BUFFER_INDEX, RO		1318
				DD	18	000C6	BGEQ	7\$		
				03	11	000C8	BRB	11\$		
		6A		01	CE	000CA	10\$:	MNEGL #1, BUFFER_INDEX		1321
				6A	D6	000CD	11\$:	INCL BUFFER_INDEX		1322
		56		6A	D0	000CF	MOVL	BUFFER_INDEX, DEL_WORD_BEGIN		1323
20	A9	6A		01	81	000D2	ADDB3	#1, BUFFER_INDEX, -AED_B_COLUMN		1324
60	A9	57		56	A3	000D7	SUBW3	DEL_WORD_BEGIN, DEL_WORD_END, -		1325
								AED_Q_DEL_WORD		
				64	A9	9F 000DC	PUSHAB	AED_Q_DEL_WORD+4		1327
				60	A9	3C 000DF	MOVZWL	AED_Q_DEL_WORD, 4(SP)		
				04	AE	9F 000E4	PUSHAB	4(SP)		
		00000000G	00	02	FB	000E7	CALLS	#2, LIB\$GET_VM		
			58	50	D0	000EE	MOVL	RO, VM_STATUS		
			08	58	E9	000F1	BLBC	VM_STATUS, 12\$		
60	A9		00	00	2C	000F4	MOVC5	#0, (SP), #0, AED_Q_DEL_WORD, -		
								2AED_Q_DEL_WORD+4		
		008C	C9	64	B9	000FA	12\$:	MOVL VM_STATUS, -AED_L_STATUS		
			57	008C	C9	E8 00101	BLBS	AED_L_STATUS, T6\$		1328
			69	03	E1	00106	BBC	#3, -AED_L_FLAGS, 13\$		1331
				01	DD	0010A	PUSHL	#1		
				15	DD	0010C	PUSHL	#21		
		00000000G	00	02	FB	0010E	CALLS	#2, SCR\$ERASE_PAGE		
				01	DD	00115	PUSHL	#1		
				15	DD	00117	PUSHL	#21		
			6B	02	FB	00119	CALLS	#2, SCR\$SET_CURSOR		
				008C	C9	DD 0011C	13\$:	PUSHL AED_L_STATUS		
		00000000G	00	01	FB	00120	CALLS	#1, -LIB\$SIGNAL		
	0B		69	03	E1	00127	BBC	#3, AED_L_FLAGS, 14\$		
			7E	20	A9	9A 0012B	MOVZBL	AED_B_COLUMN, -(SP)		
			7E	24	A9	9A 0012F	MOVZBL	AED_B_LINE, -(SP)		
			6B	02	FB	00133	CALLS	#2, SCR\$SET_CURSOR		
			50	008C	C9	D0 00136	14\$:	MOVL AED_L_STATUS, RO		
			07	50	93	0013B	BITB	RO, #7		
				11	13	0013E	BEQL	15\$		
51			03	00	EF	00140	EXTZV	#0, #3, RO, R1		
51	14	50	03	00	ED	00145	CMPZV	#0, #3, AED_L_WORSTERR, R1		
				04	18	0014B	BGEQ	15\$		
				50	D0	0014D	MOVL	RO, AED_L_WORSTERR		
		14	A9	2008	8F	AA 00151	15\$:	BICW2 #8200, AED_L_FLAGS+1		1333
		01	A9	28	AA	94 00157	CLRB	TERM_CHAR		1334

64	B9	00C4	C946	60	0085	31	0015A	BRW	20\$	1335
					A9	28	0015D	MOV C3	AED Q DEL_WORD, INPUT BUFFER-	1339
		01	A9		02	88	00166	BIS B2	[DEC WORD-BEGIN], @AED_Q_DEL_WORD+4	1340
04	AA	00B8	C9		57	A3	0016A	SUBW3	#2, AED_L_FLAGS+1	1341
		08	AA	00C4	C947	9E	00171	MOVAB	DEL_WORD_END, SEGMENT_SIZE, ECHO_DESC	1342
				04	AA	B5	00178	TSTM	INPUT_BUFFER[DEL_WORD_END], ECHO_DESC+4	1343
					13	13	0017B	BEQ L	ECHO_DESC	1346
		7E		20	A9	9A	0017D	MOVZBL	AED_B_COLUMN, -(SP)	1347
		7E		24	A9	9A	00181	MOVZBL	AED_B_LINE, -(SP)	1351
		6B			02	FB	00185	CALLS	#2, SCRSET_CURSOR	1352
				04	AA	9F	00188	PUSHAB	ECHO_DESC	1353
		0000G	CF		01	FB	0018B	CALLS	#1, AED_PUTOUTPUT	1354
50	00	00000200	8F		6A	C3	00190	SUB L3	BUFFER_INDEX, #512, R0	1355
		00C4	C947	04	AA	2C	00198	MOV C5	ECHO_DESC, INPUT_BUFFER[DEL_WORD_END], #0, -	1356
				00C4	C946		001A1		R0, INPUT_BUFFER[DEL_WORD-BEGIN]	1357
		00B8	C9	60	A9	A2	001A5	SUBW2	AED_Q_DEL_WORD, SEGMENT_SIZE	1358
			7E	00B8	C9	3C	001AB	MOVZWL	SEGMENT_SIZE, -(SP)	1359
					6E	D6	001B0	INCL	(SP)	1360
		7E		24	A9	9A	001B2	MOVZBL	AED_B_LINE, -(SP)	1361
		00000000G	00		02	FB	001B6	CALLS	#2, SCRERASE_LINE	1362
			7E	20	A9	9A	001BD	MOVZBL	AED_B_COLUMN, -(SP)	1363
			7E	24	A9	9A	001C1	MOVZBL	AED_B_LINE, -(SP)	1364
		0000G	CF		02	FB	001C5	CALLS	#2, AED_SET_CURSOR	1365
					09	11	001CA	BRB	19\$	1366
					7E	D4	001CC	CLRL	-(SP)	1367
					5A	DD	001CE	PUSHL	R10	1368
		0000G	CF		02	FB	001D0	CALLS	#2, AED_SEGCOMBINE	1369
		01	A9	2008	8F	AA	001D5	BICW2	#8200, AED_L_FLAGS+1	1370
				28	AA	94	001DB	CLRB	TERM_CHAR	1371
			50		01	D0	001DE	MOVL	#1, R0	1372
					04	04	001E1	RET		1373
					50	D4	001E2	CLRL	R0	1374
					04	04	001E4	RET		1375

; Routine Size: 485 bytes, Routine Base: \$CODE\$ + 0675

ACT_RUB_BOL - erase to beginning of line

```
920 1367 1 XSBTTL 'ACT RUB BOL - erase to beginning of line'
921 1368 1 ROUTINE ACT_RUB_BOL =
922 1369 1
923 1370 1 **
924 1371 1
925 1372 1 FUNCTIONAL DESCRIPTION:
926 1373 1
927 1374 1     This routine deletes all characters between the current cursor
928 1375 1     position and the beginning of the line segment. These characters
929 1376 1     are NOT stored.
930 1377 1
931 1378 1 CALLING SEQUENCE:
932 1379 1     ACT_RUB_BOL ()
933 1380 1
934 1381 1 INPUT PARAMETERS:
935 1382 1     none
936 1383 1
937 1384 1 IMPLICIT INPUTS:
938 1385 1     OWN storage
939 1386 1
940 1387 1 OUTPUT PARAMETERS:
941 1388 1     none
942 1389 1
943 1390 1 IMPLICIT OUTPUTS:
944 1391 1     none
945 1392 1
946 1393 1 ROUTINE VALUE:
947 1394 1     1 if successful
948 1395 1     error status otherwise
949 1396 1
950 1397 1 SIDE EFFECTS:
951 1398 1     The line segment table is updated as necessary, ACE line pointers
952 1399 1     are updated, and the object's ACL is updated as necessary.
953 1400 1
954 1401 1 --
955 1402 1
956 1403 2 BEGIN
957 1404 2
958 1405 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
959 1406 2
960 1407 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
961 1408 2 THEN
962 1409 2     BEGIN
963 1410 2         SIGNAL (AED$ NOMODIFY);
964 1411 2         AED_L_FLAGS[AED_V_GOLDKEY] = 0;
965 1412 2         AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
966 1413 2         TERM_CHAR = 0;
967 1414 2         RETURN 1;
968 1415 2     END;
969 1416 2
970 1417 2 ! If at the beginning of the line, this is a no-op.
971 1418 2
972 1419 2 IF .BUFFER_INDEX EQL 0
973 1420 2 THEN
974 1421 2     BEGIN
975 1422 2         AED_L_FLAGS[AED_V_GOLDKEY] = 0;
976 1423 2         AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
```


ACT_RUB_BOL - erase to beginning of line

```

977 1424 TERM CHAR = 0;
978 1425 RETURN 1;
979 1426 END;
980 1427
981 1428 ! Deallocate anything in the saved line buffer.
982 1429
983 1430 IF .AED_Q_DEL_LINE[DSC$W_LENGTH] NEQ 0
984 1431 THEN DEALLOCATE (.AED_Q_DEL_LINE[DSC$W_LENGTH], .AED_Q_DEL_LINE[DSC$A_POINTER]);
985 1432 AED_Q_DEL_LINE[DSC$W_LENGTH] = 0;
986 1433
987 1434 ! Delete to the beginning of the line.
988 1435
989 1436 AED_L_FLAGS[AED_V_DELBOL] = 1; ! Note direction of delete
990 1437
991 1438 AED_Q_DEL_LINE[DSC$W_LENGTH] = .BUFFER_INDEX;
992 1439 AED_L_STATUS = ALLOCATE (.AED_Q_DEL_LINE[DSC$W_LENGTH], AED_Q_DEL_LINE[DSC$A_POINTER]);
993 1440 IF NOT .AED_L_STATUS
994 1441 THEN
995 1442 BEGIN
996 1443 SIGNAL (.AED_L_STATUS);
997 1444 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
998 1445 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
999 1446 RETURN 0;
1000 1447 END;
1001 1448
1002 1449 ! Copy the deleted portion of the line.
1003 1450
1004 1451 CHSMOVE (.AED_Q_DEL_LINE[DSC$W_LENGTH], INPUT_BUFFER[0], .AED_Q_DEL_LINE[DSC$A_POINTER]);
1005 1452 SEGMENT_SIZE = .SEGMENT_SIZE - .BUFFER_INDEX;
1006 1453 CHSCOPY (.SEGMENT_SIZE, BUFFER_CHAR, 0, 512, INPUT_BUFFER);
1007 1454 BUFFER_INDEX = 0;
1008 1455
1009 1456 ! Echo any remaining portion of the line.
1010 1457
1011 1458 IF .SEGMENT_SIZE GTR 0
1012 1459 THEN
1013 1460 BEGIN
1014 1461 ECHO_DESC[DSC$W_LENGTH] = .SEGMENT_SIZE;
1015 1462 ECHO_DESC[DSC$A_POINTER] = BUFFER_CHAR;
1016 1463 SCR$SET CURSOR (.AED_B_LINE, 1);
1017 1464 AED_PUTOUTPUT (ECHO_DESC);
1018 1465 END;
1019 1466
1020 1467 ! Now clear the rest of the line.
1021 1468
1022 1469 SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
1023 1470 AED_SET_CURSOR (.AED_B_LINE, 1);
1024 1471
1025 1472 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1026 1473 AED_L_FLAGS[AED_V_MODIFIED] = 1;
1027 1474 AED_B_COLUMN = .BUFFER_INDEX + 1;
1028 1475 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1029 1476 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1030 1477 TERM CHAR = 0;
1031 1478 RETURN 1;
1032 1479
1033 1480 END; ! End of routine ACT_RUB_BOL
```

OFFC 00000 ACT_RUB_BOL:

5B	00000000G	8F	D0	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	1368
5A	00000000G	00	9E	00009	MOVL	#AED\$ NOMODIFY, R11	
59	00000000G	00	9E	00010	MOVAB	SCR\$ERASE_PAGE, R10	
58	0000'	CF	9E	00017	MOVAB	SCR\$SET_CURSOR, R9	
57	0000'	CF	9E	0001C	MOVAB	BUFFER_INDEX, R8	
5E		04	C2	00021	MOVAB	AED_L_FLAGS, R7	
50	40	A7	D0	00024	SUBL2	#4, SP	
45	0A	04	E1	00028	MOVL	AED_L_FIRSTLINE, R0	1407
0E		03	E1	0002D	BBC	#4, 10(R0), 4\$	
		01	DD	00031	BBC	#3, AED_L_FLAGS, 1\$	1410
		15	DD	00033	PUSHL	#1	
6A		02	FB	00035	PUSHL	#21	
		01	DD	00038	CALLS	#2, SCR\$ERASE_PAGE	
		15	DD	0003A	PUSHL	#1	
69		02	FB	0003C	PUSHL	#21	
		5B	DD	0003F	CALLS	#2, SCR\$SET_CURSOR	
0B	00000000G	00	01	FB	PUSHL	R11	
		67	03	E1	CALLS	#1, LIB\$SIGNAL	
		7E	A7	9A	BBC	#3, AED_L_FLAGS, 2\$	
		7E	A7	9A	MOVZBL	AED_B_COLUMN, -(SP)	
		69	02	FB	MOVZBL	AED_B_LINE, -(SP)	
			8F	D5	CALLS	#2, SCR\$SET_CURSOR	
00000000*			10	13	TSTL	#<AED\$ NOMODIFY&7>	
		03	00	ED	BEQL	3\$	
			04	18	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$ NOMODIFY&7>	
		14	A7	5B	BGEQ	3\$	
			0111	31	MOVL	R11, AED_L_WORSTERR	
			68	D5	BRW	12\$	1411
			F9	13	TSTL	BUFFER_INDEX	1419
		50	58	A7	BEQL	3\$	
			11	13	MOVZWL	AED_Q_DEL_LINE, R0	1430
			5C	A7	BEQL	5\$	
		04	AE	50	PUSHL	AED_Q_DEL_LINE+4	1431
			04	AE	MOVL	R0, 4(SP)	
00000000G	00		02	FB	PUSHAB	4(SP)	
		58	A7	B4	CALLS	#2, LIB\$FREE VM	
			04	88	CLRW	AED_Q_DEL_LINE	1432
01	A7		68	B0	BISB2	#4, AED_L_FLAGS+1	1436
58	A7		A7	9F	MOVW	BUFFER_INDEX, AED_Q_DEL_LINE	1438
		5C	A7	3C	PUSHAB	AED_Q_DEL_LINE+4	1439
		58	A7	3C	MOVZWL	AED_Q_DEL_LINE, 4(SP)	
		04	AE	9F	PUSHAB	4(SP)	
00000000G	00		02	FB	CALLS	#2, LIB\$GET VM	
			50	D0	MOVL	R0, VM_STATUS	
			56	E9	BLBC	VM_STATUS, 6\$	
5B	A7		00	2C	MOVCS	#0, (SP), #0, AED_Q_DEL_LINE, -	
		5C	B7			0AED_Q_DEL_LINE+4	
	008C	C7	56	D0	MOVL	VM_STATUS, AED_L_STATUS	1440
		4F	C7	E8	BLBS	AED_L_STATUS, 10\$	
0E		67	03	E1	BBC	#3, AED_L_FLAGS, 7\$	1443
			01	DD	PUSHL	#1	

51	51	14	50	A7	03	00	EF	000F8	EXTZV	#0, #3, R0, R1			
					03	00	ED	000FD	CMPZV	#0, #3, AED_L_WORSTERR, R1			
			14	A7	50	D0	00105	BGEQ	9\$				
			01	A7	2008	8F	AA	00109	9\$:	BICW2	#8200, AED_L_FLAGS+1	1445	
						7F	11	0010F	BRB	13\$		1446	
		5C	B7	00C4	C7	58	A7	28	00111	10\$:	MOVCS	AED Q DEL LINE, INPUT_BUFFER, -	1451
				00B8	C7	68	A2	00119	SUBW2	2AED Q DEL LINE+4			
				50	00C4	C7	9E	0011E	MOVAB	BUFFER_INDEX, SEGMENT_SIZE		1452	
0200	8F		00	00	00B8	C7	2C	00123	MOVCS	INPUT_BUFFER, R0		1453	
					00C4	C7		0012E		SEGMENT_SIZE, @BUFFER_INDEX[R0], #0, #512, -			
						68	D4	00131	CLRL	INPUT_BUFFER		1454	
				50	00B8	C7	3C	00133	MOVZWL	BUFFER_INDEX		1458	
						1F	15	00138	BLEQ	SEGMENT_SIZE, R0			
			04	A8	50	B0	0013A	MOVW	11\$				
				50	00C4	C7	9E	0013E	MOVW	R0, ECHO_DESC		1461	
		08	A8	50		68	C1	00143	MOVAB	INPUT_BUFFER, R0		1462	
						01	DD	00148	ADDL3	BUFFER_INDEX, R0, ECHO_DESC+4			
				7E	24	A7	9A	0014A	PUSHL	#1		1463	
				69	04	A8	9F	00151	MOVZBL	AED_B LINE, -(SP)			
						02	FB	0014E	CALLS	#2, -SCRSET_CURSOR			
				0000G	CF	01	FB	00154	PUSHAB	ECHO_DESC		1464	
				7E	00B8	C7	3C	00159	CALLS	#1, AED_PUTOUTPUT		1469	
						6E	D6	0015E	MOVZWL	SEGMENT_SIZE, -(SP)			
				7E	24	A7	9A	00160	INCL	(SP)			
				00000000G	00	02	FB	00164	MOVZBL	AED_B LINE, -(SP)			
						01	DD	00168	CALLS	#2, -SCRERASE_LINE			
				7E	24	A7	9A	0016D	PUSHL	#1		1470	
				0000G	CF	02	FB	00171	MOVZBL	AED_B LINE, -(SP)			
				01	A7	10	8A	00176	CALLS	#2, -AED SET_CURSOR			
				67	80	8F	88	0017A	BICB2	#16, AED_L_FLAGS+1		1472	
				68	01	81	0017E	BISB2	#128, AED_L_FLAGS			1473	
20	A7			01	A7	2008	8F	AA	ADDL3	#1, BUFFER_INDEX, AED_B_COLUMN		1474	
						28	A8	94	BICW2	#8200, AED_L_FLAGS+1		1476	
				50		01	D0	0018C	CLRB	TERM_CHAR		1477	
							04	0018F	MOVL	#1, R0		1478	
						50	D4	00190	RET				
						04	00192	CLRL	R0			1480	
									RET				

; Routine Size: 403 bytes, Routine Base: \$CODE\$ + 085A

ACT_DEL_CHR - delete current character

```
1035 1481 1 XSBTTL 'ACT_DEL_CHR - delete current character'
1036 1482 1 ROUTINE ACT_DEL_CHR =
1037 1483 1
1038 1484 1 **
1039 1485 1
1040 1486 1 FUNCTIONAL DESCRIPTION:
1041 1487 1
1042 1488 1 This routine deletes the character immediately under the current
1043 1489 1 cursor position. The deleted character is placed in storage for
1044 1490 1 later retrieval.
1045 1491 1
1046 1492 1 CALLING SEQUENCE:
1047 1493 1 ACT_DEL_CHR ()
1048 1494 1
1049 1495 1 INPUT PARAMETERS:
1050 1496 1 none
1051 1497 1
1052 1498 1 IMPLICIT INPUTS:
1053 1499 1 OWN storage
1054 1500 1
1055 1501 1 OUTPUT PARAMETERS:
1056 1502 1 none
1057 1503 1
1058 1504 1 IMPLICIT OUTPUTS:
1059 1505 1 none
1060 1506 1
1061 1507 1 ROUTINE VALUE:
1062 1508 1 1 if successful
1063 1509 1 error status otherwise
1064 1510 1
1065 1511 1 SIDE EFFECTS:
1066 1512 1 The line segment table is updated as necessary, ACE line pointers
1067 1513 1 are updated, and the object's ACL is updated as necessary.
1068 1514 1
1069 1515 1 --
1070 1516 1
1071 1517 1 BEGIN
1072 1518 1
1073 1519 1 ! If the current ACE is marked as untouchable, no modifications are allowed.
1074 1520 1
1075 1521 1 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1076 1522 1 THEN
1077 1523 1 BEGIN
1078 1524 1 SIGNAL (AED$ NOMODIFY);
1079 1525 1 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1080 1526 1 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1081 1527 1 TERM CHAR = 0;
1082 1528 1 RETURN 1;
1083 1529 1 END;
1084 1530 1
1085 1531 1 ! Delete the character.
1086 1532 1
1087 1533 1 AED_B_DEL_CHAR = 0;
1088 1534 1
1089 1535 1 AED_L_FLAGS[AED_V_MODIFIED] = 1;
1090 1536 1 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1091 1537 1
```


: 1525

03

		68	A6	94	0006C	4\$:	CLRB	AED_B_DEL_CHAR	1533
		80	8F	88	0006F		BISB2	#128, AED_L_FLAGS	1535
01	66		10	8A	00073		BICB2	#16, AED_L_FLAGS+1	1536
	A6	00B8	C6	3C	00077		MOVZWL	SEGMENT_SIZE, R1	1538
	51		71	15	0007C		BLEQ	7\$	
	51		67	D1	0007E		CMPL	BUFFER_INDEX, R1	1539
			6C	18	00081		BGEQ	7\$	
	50		67	D0	00083		MOVL	BUFFER_INDEX, R0	1542
68	A6	00C4	C640	90	00086		MOVB	INPUT_BUFFER[R0], AED_B_DEL_CHAR	
5D	8F	68	A6	91	0008D		CMPB	AED_B_DEL_CHAR, #93	1543
			04	12	00092		BNEQ	5\$	
02	A6		01	88	00094		BISB2	#1, AED_L_FLAGS+2	
	51		50	C2	00098	5\$:	SUBL2	R0, R1	1544
04	A7		01	A3	0009B		SUBW3	#1, R1, ECHO_DESC	
08	A7	00C5	C640	9E	000A0		MOVAB	INPUT_BUFFER+1[R0], ECHO_DESC+4	1545
		04	A7	B5	000A7		TSTW	ECHO_DESC	1546
			08	13	000AA		BEQL	6\$	
		04	A7	9F	000AC		PUSHAB	ECHO_DESC	1547
0000G	CF		01	FB	000AF		CALLS	#1, AED_PUTOUTPUT	
	7E	00B8	C6	3C	000B4	6\$:	MOVZWL	SEGMENT_SIZE, -(SP)	1548
	7E	24	A6	9A	000B9		MOVZBL	AED_B_LINE, -(SP)	
00000000G	00		02	FB	000BD		CALLS	#2, SCRSErase_LINE	
	7E	20	A6	9A	000C4		MOVZBL	AED_B_COLUMN, -(SP)	1549
	7E	24	A6	9A	000C8		MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	000CC		CALLS	#2, AED_SET_CURSOR	
	50		67	D0	000D1		MOVL	BUFFER_INDEX, R0	1550
51	00000200		50	C3	000D4		SUBL3	R0, #512, R1	1552
00	00C5	C640	04	A7	2C	000DC	MOVCS	ECHO_DESC, INPUT_BUFFER+1[R0], #0, R1, -	
			00C4	C640		000E5		INPUT_BUFFER[R0]	
		00B8	C6	B7	000E9		DECW	SEGMENT_SIZE	1553
			09	11	000ED		BRB	8\$	1538
			01	DD	000EF	7\$:	PUSHL	#1	1555
			57	DD	000F1		PUSHL	R7	
0000G	CF		02	FB	000F3		CALLS	#2, AED_SEGCOMBINE	
01	A6	2008	8F	AA	000F8	8\$:	BICW2	#8200, AED_L_FLAGS+1	1558
		28	A7	94	000FE		CLRB	TERM_CHAR	1559
	50		01	D0	00101		MOVL	#1, R0	1560
			04	04	00104		RET		1562

; Routine Size: 261 bytes, Routine Base: \$CODE\$ + 09ED

ACT_DEL_WRD - delete current word

```
1118 1563 1 XSBTTL 'ACT_DEL_WRD - delete current word'
1119 1564 1 ROUTINE ACT_DEL_WRD =
1120 1565 1
1121 1566 1 ++
1122 1567 1
1123 1568 1 FUNCTIONAL DESCRIPTION:
1124 1569 1
1125 1570 1     This routine deletes the word (all characters until the first non
1126 1571 1     alphanumeric character) starting at the current cursor position.
1127 1572 1     The deleted word is placed in storage for later retrieval.
1128 1573 1
1129 1574 1 CALLING SEQUENCE:
1130 1575 1     ACT_DEL_WRD ()
1131 1576 1
1132 1577 1 INPUT PARAMETERS:
1133 1578 1     none
1134 1579 1
1135 1580 1 IMPLICIT INPUTS:
1136 1581 1     OWN storage
1137 1582 1
1138 1583 1 OUTPUT PARAMETERS:
1139 1584 1     none
1140 1585 1
1141 1586 1 IMPLICIT OUTPUTS:
1142 1587 1     none
1143 1588 1
1144 1589 1 ROUTINE VALUE:
1145 1590 1     1 if successful
1146 1591 1     error status otherwise
1147 1592 1
1148 1593 1 SIDE EFFECTS:
1149 1594 1     The line segment table is updated as necessary, ACE line pointers
1150 1595 1     are updated, and the object's ACL is updated as necessary.
1151 1596 1
1152 1597 1 --
1153 1598 1
1154 1599 2 BEGIN
1155 1600 2
1156 1601 2 LOCAL
1157 1602 2     DEL_WORD_BEGIN,           ! Beginning offset of word
1158 1603 2     DEL_WORD_END;             ! End offset of word
1159 1604 2
1160 1605 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
1161 1606 2
1162 1607 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1163 1608 2 THEN
1164 1609 2     BEGIN
1165 1610 2         SIGNAL (AED$ NOMODIFY);
1166 1611 2         AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1167 1612 2         AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1168 1613 2         TERM CHAR = 0;
1169 1614 2         RETURN 1;
1170 1615 2     END;
1171 1616 2
1172 1617 2 ! Deallocate anything in the saved word buffer.
1173 1618 2
1174 1619 2 IF .AED_O_DEL_WORD[DSC$W_LENGTH] NEQ 0
```

ACT_DEL_WRD - delete current word

```
1175 1620 2 THEN DEALLOCATE (.AED_Q_DEL_WORD[DSC$W_LENGTH], AED_Q_DEL_WORD[DSC$A_POINTER]);
1176 1621 2 AED_Q_DEL_WORD[DSC$W_LENGTH] = 0;
1177 1622 2
1178 1623 2 AED_L_FLAGS[AED_V_MODIFIED] = 1;
1179 1624 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1180 1625 2
1181 1626 2 ! Delete the word.
1182 1627 2
1183 1628 2 IF .SEGMENT_SIZE GTR 0
1184 1629 2 AND .BUFFER_INDEX LSS .SEGMENT_SIZE
1185 1630 2 THEN
1186 1631 2 BEGIN
1187 1632 2 DEL_WORD_BEGIN = .BUFFER_INDEX;
1188 1633 2 WHILE (.BUFFER_CHAR GEQ 'A' AND .BUFFER_CHAR LEQ 'Z')
1189 1634 2 OR (.BUFFER_CHAR GEQ '0' AND .BUFFER_CHAR LEQ '9')
1190 1635 2 DO
1191 1636 2 BEGIN
1192 1637 2 BUFFER_INDEX = .BUFFER_INDEX + 1;
1193 1638 2 IF .BUFFER_INDEX GEQ .SEGMENT_SIZE
1194 1639 2 THEN
1195 1640 2 BEGIN
1196 1641 2 BUFFER_INDEX = .BUFFER_INDEX - 1;
1197 1642 2 EXITLOOP;
1198 1643 2 END;
1199 1644 2 END;
1200 1645 2 BUFFER_INDEX = .BUFFER_INDEX + 1; ! First char past delimiter
1201 1646 2 DEL_WORD_END = .BUFFER_INDEX;
1202 1647 2 AED_Q_DEL_WORD[DSC$W_LENGTH] = .DEL_WORD_END - .DEL_WORD_BEGIN;
1203 1648 2 AED_L_STATUS = ALLOCATE (.AED_Q_DEL_WORD[DSC$W_LENGTH],
P 1649 2 AED_Q_DEL_WORD[DSC$A_POINTER]);
1204 1650 2
1205 1651 2 IF NOT .AED_L_STATUS
1206 1652 2 THEN
1207 1653 2 BEGIN
1208 1654 2 SIGNAL (.AED_L_STATUS);
1209 1655 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1210 1656 2 RETURN 0;
1211 1657 2 END;
1212 1658 2 CH$MOVE (.AED_Q_DEL_WORD[DSC$W_LENGTH],
1213 1659 2 INPUT_BUFFER[DEL_WORD_BEGIN],
1214 1660 2 .AED_Q_DEL_WORD[DSC$A_POINTER]);
1215 1661 2 AED_L_FLAGS[AED_V_ROBWORD] = 0;
1216 1662 2 ECHO_DESC[DSC$W_LENGTH] = .SEGMENT_SIZE - .DEL_WORD_END;
1217 1663 2 ECHO_DESC[DSC$A_POINTER] = INPUT_BUFFER[DEL_WORD_END];
1218 1664 2 SEGMENT_SIZE = .SEGMENT_SIZE - .AED_Q_DEL_WORD[DSC$W_LENGTH];
1219 1665 2 IF .ECHO_DESC[DSC$W_LENGTH] GEQ 1
1220 1666 2 THEN
1221 1667 2 BEGIN
1222 1668 2 SCR$SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1223 1669 2 AED_PUTOUTPUT (ECHO_DESC);
1224 1670 2 END;
1225 1671 2 CH$COPY (.ECHO_DESC[DSC$W_LENGTH], INPUT_BUFFER[DEL_WORD_END],
1226 1672 2 0,
1227 1673 2 512 - .BUFFER_INDEX, INPUT_BUFFER[DEL_WORD_BEGIN]);
1228 1674 2 BUFFER_INDEX = .DEL_WORD_BEGIN;
1229 1675 2 SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
1230 1676 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1231 1677 2 END
```



```
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1677 2 ELSE AED_SEGCOMBINE (BUFFER_INDEX, 1);  
1678 2  
1679 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;  
1680 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;  
1681 2 TERM_CHAR = 0;  
1682 2 RETURN 1;  
1683 2  
1684 1 END;
```

! End of routine ACT_DEL_WRD

				OFFC 00000 ACT_DEL_WRD:			
			5B 00000000G	00 9E 00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	1564
			5A 0000'	CF 9E 00009	MOVAB	SCR\$SET CURSOR, R11	
			59 0000'	CF 9E 0000E	MOVAB	BUFFER_INDEX, R10	
			5E	04 C2 00013	MOVAB	AED_L_FLAGS, R9	
			50 40	A9 D0 00016	SUBL2	#4, -SP	
51	0A		A0	04 E1 0001A	MOVL	AED_L_FIRSTLINE, R0	1607
12			69	03 E1 0001F	BBC	#4, -10(R0), 4\$	
				01 DD 00023	BBC	#3, AED_L_FLAGS, 1\$	1610
				15 DD 00025	PUSHL	#1	
		00000000G	00	02 FB 00027	PUSHL	#21	
				01 DD 0002E	CALLS	#2, SCR\$ERASE_PAGE	
				15 DD 00030	PUSHL	#1	
			6B	02 FB 00032	PUSHL	#21	
		00000000G	00	8F DD 00035 1\$:	CALLS	#2, SCR\$SET CURSOR	
0B			69	01 FB 0003B	PUSHL	#AED\$ NOMODIFY	
			7E 20	03 E1 00042	CALLS	#1, LIB\$SIGNAL	
			7E 24	A9 9A 00046	BBC	#3, AED_L_FLAGS, 2\$	
			6B	A9 9A 0004A	MOVZBL	AED_B_COLUMN, -(SP)	
				02 FB 0004E	MOVZBL	AED_B_LINE, -(SP)	
		00000000*		8F D5 00051 2\$:	CALLS	#2, SCR\$SET CURSOR	
				14 13 00057	TSTL	#<AED\$ NOMODIFY&7>	
00000000*	8F	14	A9	00 ED 00059	BEQL	3\$	
				08 18 00063	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$ NOMODIFY&7>	
		14	A9 00000000G	8F D0 00065	BGEQ	3\$	
				015E 31 0006D 3\$:	MOVL	#AED\$ NOMODIFY, AED_L_WORSTERR	
			50 60	A9 3C 00070 4\$:	BRW	19\$	1611
				11 13 00074	MOVZWL	AED_Q_DEL_WORD, R0	1619
				A9 9F 00076	BEQL	5\$	
		04	AE	50 D0 00079	PUSHAB	AED_Q_DEL_WORD+4	1620
				04 AE 9F 0007D	MOVL	R0, -4(SP)	
		00000000G	00	02 FB 00080	PUSHAB	4(SP)	
				60 A9 B4 00087 5\$:	CALLS	#2, LIB\$FREE VM	
			69 80	8F 88 0008A	CLRW	AED_Q_DEL_WORD	1621
				10 8A 0008E	BISB2	#128, AED_L_FLAGS	1623
01			A9	C9 3C 00092	BICB2	#16, AED_L_FLAGS+1	1624
			51 00B8	03 14 00097	MOVZWL	SEGMENT_SIZE, R1	1628
				0129 31 00099 6\$:	BGTR	7\$	
			51	6A D1 0009C 7\$:	BRW	18\$	
				F8 18 0009F	CNPL	BUFFER_INDEX, R1	1629
			57	6A D0 000A1	BGEQ	6\$	
			50 00C4	C9 9E 000A4 8\$:	MOVL	BUFFER_INDEX, DEL_WORD_BEGIN	1632
			50	00 BA40 9A 000A9	MOVAB	INPUT_BUFFER, R0	1633
					MOVZBL	@BUFFER_INDEX[R0], R0	

		41	8F		50	91	000AE		CMPB	R0,	#65		
		5A	8F		06	1F	000B2		BLSSU	9\$			
					50	91	000B4		CMPB	R0,	#90		
			30		0A	1B	000B3		BLEQU	10\$			
					50	91	000EA	9\$:	CMPB	R0,	#48		1634
			39		0E	1F	000BD		BLSSU	11\$			
					50	91	000BF		CMPB	R0,	#57		
					09	1A	000C2		BGTRU	11\$			
					6A	D6	000C4	10\$:	INCL	BUFFER_INDEX			1637
			51		6A	D1	000C6		CMPL	BUFFER_INDEX, R1			1638
					D9	19	000C9		BLSS	8\$			
					6A	D7	000CB		DECL	BUFFER_INDEX			1641
					6A	D6	000CD	11\$:	INCL	BUFFER_INDEX			1645
			56		6A	D0	000CF		MOVL	BUFFER_INDEX, DEL_WORD_END			1646
60	A9		56		57	A3	000D2		SUBW3	DEL_WORD_BEGIN, DEL_WORD_END, -			1647
										AED_Q_DEL_WORD			
				64	A9	9F	000D7		PUSHAB	AED_Q_DEL_WORD+4			1649
		04	AE	60	A9	3C	000DA		MOVZWL	AED_Q_DEL_WORD, 4(SP)			
				04	AE	9F	000DF		PUSHAB	4(SP)			
		00000000G	00		02	FB	000E2		CALLS	#2, LIB\$GET_VM			
			58		50	D0	000E9		MOVL	R0, VM_STATUS			
			08		58	E9	000EC		BLBC	VM_STATUS, 12\$			
60	A9		6E		00	2C	000EF		MOVCS	#0, (SP), #0, AED_Q_DEL_WORD, -			
				64	B9		000F5			AED_Q_DEL_WORD+4			
		008C	C9		58	D0	000F7	12\$:	MOVL	VM_STATUS, AED_L_STATUS			
			52	008C	C9	E8	000FC		BLBS	AED_L_STATUS, 16\$			1650
			69		03	E1	00101		BBC	#3, AED_L_FLAGS, 13\$			1653
					01	DD	00105		PUSHL	#1			
					15	DD	00107		PUSHL	#21			
		00000000G	00		02	FB	00109		CALLS	#2, SCR\$ERASE_PAGE			
					01	DD	00110		PUSHL	#1			
					15	DD	00112		PUSHL	#21			
			6B		02	FB	00114		CALLS	#2, SCR\$SET_CURSOR			
		00000000G	00	008C	C9	DD	00117	13\$:	PUSHL	AED_L_STATUS			
			69		01	FB	0011B		CALLS	#1, LIB\$SIGNAL			
		08	7E		03	E1	00122		BBC	#3, AED_L_FLAGS, 14\$			
			7E	20	A9	9A	00126		MOVZBL	AED_B_COLUMN, -(SP)			
			6B	24	A9	9A	0012A		MOVZBL	AED_B_LINE, -(SP)			
			50		02	FB	0012E		CALLS	#2, SCR\$SET_CURSOR			
			07	008C	C9	D0	00131	14\$:	MOVL	AED_L_STATUS, R0			
					50	93	00136		BITB	R0, #7			
					11	13	00139		BEQL	15\$			
51			03		00	EF	0013B		EXTZV	#0, #3, R0, R1			
51	14	50	03		00	ED	00140		CMPZV	#0, #3, AED_L_WORSTERR, R1			
					04	18	00146		BGEQ	15\$			
		14	A9		50	D0	00148		MOVL	R0, AED_L_WORSTERR			
		01	A9		08	8A	0014C	15\$:	BICB2	#8, AED_L_FLAGS+1			1654
					0088	31	00150		BRW	20\$			1655
		64	B9	00C4	C947	60	A9	28	00153	16\$:	MOVC3	AED_Q_DEL_WORD, INPUT_BUFFER-	1659
										[DEL_WORD_BEGIN], AED_Q_DEL_WORD+4			
		01	A9		02	8A	0015C		BICB2	#2, AED_L_FLAGS+1			1660
		04	AA	00B8	C9	56	A3	00160	SUBW3	DEL_WORD_END, SEGMENT_SIZE, ECHO_DESC			1661
									MOVAB	INPUT_BUFFER[DEL_WORD_END], ECHO_DESC+4			1662
					00C4	C946	9E	00167	SUBW2	AED_Q_DEL_WORD, SEGMENT_SIZE			1663
					60	A9	A2	0016E	TSTW	ECHO_DESC			1664
					04	AA	B5	00174	BEQL	17\$			
									MOVZBL	AED_B_COLUMN, -(SP)			1667
			7E	20	A9	9A	00179						

50	00000000G	CF	7E	24	A9	9A	0017D	MOVZBL	AED_B_LINE, -(SP)	
			6B		02	FB	00181	CALLS	#2, SCR\$SET_CURSOR	
				04	AA	9F	00184	PUSHAB	ECHO_DESC	1668
	000000200	8F	01		FB	00187	CALLS	#1, AED_PUTOUTPUT		
	00C4 C946		6A		C3	0018C	SUBL3	BUFFER_INDEX, #512, R0		1672
			AA	04	2C	00194	MOVCS	ECHO_DESC, INPUT_BUFFER[DEL_WORD-END], #0, -		
				00C4 C947				R0, INPUT_BUFFER[DEL_WORD-BEGIN]-		
			6A		57	DD	001A1	MOVL	DEL_WORD_BEGIN, BUFFER_INDEX	1673
			7E	00B8	C9	3C	001A4	MOVZWL	SEGMENT_SIZE, -(SP)	1674
					6E	D6	001A9	INCL	(SP)	
	00000000G	00	7E	24	A9	9A	001AB	MOVZBL	AED_B_LINE, -(SP)	
			7E	20	02	FB	001AF	CALLS	#2, SCR\$ERASE_LINE	
			7E	24	A9	9A	001B6	MOVZBL	AED_B_COLUMN, -(SP)	1675
	0000G	CF			A9	9A	001BA	MOVZBL	AED_B_LINE, -(SP)	
					02	FB	001BE	CALLS	#2, AED_SET_CURSOR	
					09	11	001C3	BRB	19\$	1628
					01	DD	001C5	PUSHL	#1	1677
	0000G	CF			5A	DD	001C7	PUSHL	R10	
	01	A9			02	FB	001C9	CALLS	#2, AED_SEGCOMBINE	
				2008	8F	AA	001CE	BICW2	#8200, AED_L_FLAGS+1	1680
				28	AA	94	001D4	CLRB	TERM_CHAR	1681
			50		01	DD	001D7	MOVL	#1, R0	1682
						04	001DA	RET		
					50	D4	001DB	CLRL	R0	1684
					04	001DD	RET			

; Routine Size: 478 bytes, Routine Base: \$CODE\$ + 0AF2

ACT_DEL_EOL - delete to end of line

```
1241 1685 1 %SBTTL 'ACT DEL EOL - delete to end of line'
1242 1686 1 ROUTINE ACT_DEL_EOL =
1243 1687 1
1244 1688 1 ++
1245 1689 1
1246 1690 1 FUNCTIONAL DESCRIPTION:
1247 1691 1
1248 1692 1     This routine deletes from the current position in the line to the
1249 1693 1     end of the current line segment.
1250 1694 1
1251 1695 1 CALLING SEQUENCE:
1252 1696 1     ACT_DEL_EOL ()
1253 1697 1
1254 1698 1 INPUT PARAMETERS:
1255 1699 1     none
1256 1700 1
1257 1701 1 IMPLICIT INPUTS:
1258 1702 1     OWN storage
1259 1703 1
1260 1704 1 OUTPUT PARAMETERS:
1261 1705 1     none
1262 1706 1
1263 1707 1 IMPLICIT OUTPUTS:
1264 1708 1     none
1265 1709 1
1266 1710 1 ROUTINE VALUE:
1267 1711 1     1 if successful
1268 1712 1     error status otherwise
1269 1713 1
1270 1714 1 SIDE EFFECTS:
1271 1715 1     The line segment table is updated as necessary.
1272 1716 1
1273 1717 1 --
1274 1718 1
1275 1719 2 BEGIN
1276 1720 2
1277 1721 2 ! Check to see if the ACE is untouchable. If so, it cannot be modified.
1278 1722 2
1279 1723 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1280 1724 2 THEN
1281 1725 2     BEGIN
1282 1726 2         SIGNAL (AED$ NOMODIFY);
1283 1727 2         AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1284 1728 2         AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1285 1729 2         TERM CHAR = 0;
1286 1730 2         RETURN 1;
1287 1731 2     END;
1288 1732 2
1289 1733 2 ! Delete anything currently in the saved line buffer.
1290 1734 2
1291 1735 2 IF .AED_Q_DEL_LINE[DSCSW_LENGTH] NEQ 0
1292 1736 2 THEN DEALLOCATE (.AED_Q_DEL_LINE[DSCSW_LENGTH], .AED_Q_DEL_LINE[DSCSA_POINTER]);
1293 1737 2 AED_Q_DEL_LINE[DSCSW_LENGTH] = 0;
1294 1738 2
1295 1739 2 ! Note that the line has been modified.
1296 1740 2
1297 1741 2 AED_L_FLAGS[AED_V_MODIFIED] = 1;
```


ACT_DEL_EOL - delete to end of line

```
1298 1742 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1299 1743
1300 1744 ! Delete to the end of the line.
1301 1745
1302 1746 AED_L_FLAGS[AED_V_DELBOL] = 0; ! Note direction of delete
1303 1747
1304 1748 IF .SEGMENT_SIZE GTR 0
1305 1749 AND .BUFFER_INDEX LSS .SEGMENT_SIZE
1306 1750 THEN
1307 1751 BEGIN
1308 P 1752 AED_L_STATUS = ALLOCATE ((.SEGMENT_SIZE - .BUFFER_INDEX),
1309 1753 AED_Q_DEL_LINE[DSC$A_POINTER]);
1310 1754
1311 1755 IF NOT .AED_L_STATUS
1312 1756 THEN
1313 1757 BEGIN
1314 1758 SIGNAL (.AED_L_STATUS);
1315 1759 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1316 1760 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1317 1761 TERM_CHAR = 0;
1318 1762 RETURN 0;
1319 1763 END;
1320 1764 AED_Q_DEL_LINE[DSC$W_LENGTH] = .SEGMENT_SIZE - .BUFFER_INDEX;
1321 1765 CH$MOVE (.AED_Q_DEL_LINE[DSC$W_LENGTH], INPUT_BUFFER[.BUFFER_INDEX],
1322 1766 .AED_Q_DEL_LINE[DSC$A_POINTER]);
1323 1767 SEGMENT_SIZE = .BUFFER_INDEX;
1324 1768 SCR$ERASE_LINE (.AED_B_LINE, .AED_B_COLUMN);
1325 1769 END
1326 1770 ELSE AED_SEGCOMBINE (BUFFER_INDEX, 1);
1327 1771
1328 1772 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1329 1773 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1330 1774 TERM_CHAR = 0;
1331 1775 RETURN 1;
1332 1776 1 END;
```

OFFC 00000 ACT_DEL_EOL:

5B	00000000G	8F	D0	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	1686
5A	00000000G	00	9E	00009	MOVL	#AED\$ NOMODIFY, R11	
59	0000'	CF	9E	00010	MOVAB	SCR\$ERASE_PAGE, R10	
58	00000000G	00	9E	00015	MOVAB	BUFFER_INDEX, R9	
57	0000'	CF	9E	0001C	MOVAB	SCR\$SET_CURSOR, R8	
5E		04	C2	00021	SUBL2	#4, -SP	
50	40	A7	D0	00024	MOVL	AED_L_FIRSTLINE, R0	1723
45	0A	04	E1	00028	BBC	#4, -10(R0), 4\$	
0E		03	E1	0002D	BBC	#3, AED_L_FLAGS, 1\$	1726
		01	DD	00031	PUSHL	#1	
		15	DD	00033	PUSHL	#21	
6A		02	FB	00035	CALLS	#2, SCR\$ERASE_PAGE	
		01	DD	00038	PUSHL	#1	
		15	DD	0003A	PUSHL	#21	
68		02	FB	0003C	CALLS	#2, SCR\$SET_CURSOR	

00000000*	8F	14	A7	03	00	5B	DD	0003F	1\$:	PUSHL	R11			
					0B	01	FB	00041		CALLS	#1, LIB\$SIGNAL			
						03	E1	00048		BBC	#3, AED_L_FLAGS, 2\$			
						07	9A	0004C		MOVZBL	AED_B_COLUMN, -(SP)			
						07	9A	00050		MOVZBL	AED_B_LINE, -(SP)			
						02	FB	00054		CALLS	#2, SCR\$SET_CURSOR			
						08	D5	00057	2\$:	TSTL	#<AED\$_NOMODIFY&7>			
						10	13	0005D		BEQL	3\$			
						00	ED	0005F		CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOMODIFY&7>			
						04	18	00069		BGEQ	3\$			
						5B	D0	0006B		MOVL	R11, AED_L_WORSTERR			
						00E6	31	0006F	3\$:	BRW	14\$		1727	
						50	A7	3C	00072	4\$:	MOVZWL	AED_Q_DEL_LINE, R0	1735	
						11	13	00076		BEQL	5\$			
						5C	A7	DD	00078		PUSHL	AED_Q_DEL_LINE+4	1736	
						50	D0	0007B		MOVL	R0, 4(TSP)			
						04	AE	9F	0007F		PUSHAB	4(SP)		
						00	02	FB	00082		CALLS	#2, LIB\$FREE_VM		
						58	A7	B4	00089	5\$:	CLRW	AED_Q_DEL_LINE	1737	
						67	8F	88	0008C		BISB2	#128, AED_L_FLAGS	1741	
						01	A7	8A	00090		BICB2	#20, AED_L_FLAGS+1	1746	
						50	C7	3C	00094		MOVZWL	SEGMENT_SIZE, R0	1748	
						03	14	00099		BGTR	7\$			
						00B1	31	0009B	6\$:	BRW	13\$			
						50	D1	0009E	7\$:	CMPZV	BUFFER_INDEX, R0		1749	
						F8	18	000A1		BGEQ	6\$			
						5C	A7	9F	000A3		PUSHAB	AED_Q_DEL_LINE+4	1753	
						69	C3	000A6		SUBL3	BUFFER_INDEX, R0, 4(SP)			
						04	AE	9F	000AB		PUSHAB	4(SP)		
						00	02	FB	000AE		CALLS	#2, LIB\$GET_VM		
						56	D0	000B5		MOVL	R0, VM_STATUS			
						0F	56	E9	000B8		BLBC	VM_STATUS, 8\$		
						50	C7	3C	000BB		MOVZWL	SEGMENT_SIZE, R0		
						50	C2	000C0		SUBL2	BUFFER_INDEX, R0			
						6E	00	2C	000C3		MOVCS	#0, (SP), #0, R0, @AED_Q_DEL_LINE+4		
						5C	B7	000C8						
						00BC	C7	56	D0	000CA	8\$:	MOVL	VM_STATUS, AED_L_STATUS	
						52	C7	E8	000CF		BLBS	AED_L_STATUS, T2\$	1754	
						67	03	E1	000D4		BBC	#3, AED_L_FLAGS, 9\$	1757	
							01	DD	000D8		PUSHL	#1		
							15	DD	000DA		PUSHL	#21		
							02	FB	000DC		CALLS	#2, SCR\$ERASE_PAGE		
							01	DD	000DF		PUSHL	#1		
							15	DD	000E1		PUSHL	#21		
							02	FB	000E3		CALLS	#2, SCR\$SET_CURSOR		
							C7	DD	000E6	9\$:	PUSHL	AED_L_STATUS		
						00	01	FB	000EA		CALLS	#1, LIB\$SIGNAL		
						0B	03	E1	000F1		BBC	#3, AED_L_FLAGS, 10\$		
							A7	9A	000F5		MOVZBL	AED_B_COLUMN, -(SP)		
							A7	9A	000F9		MOVZBL	AED_B_LINE, -(SP)		
							02	FB	000FD		CALLS	#2, SCR\$SET_CURSOR		
							C7	D0	00100	10\$:	MOVL	AED_L_STATUS, R0		
							50	93	00105		BITB	R0, #7		
							11	13	00108		BEQL	11\$		
							00	EF	0010A		EXTZV	#0, #3, R0, R1		
							00	ED	0010F		CMPZV	#0, #3, AED_L_WORSTERR, R1		
							04	18	00115		BGEQ	11\$		

; Routine Size: 360 bytes, Routine Base: SCODES + 0C00

ACT_DEL_ACE - delete current ACE

```
1334 1777 1 XSBTTL 'ACT_DEL_ACE - delete current ACE'
1335 1778 1 ROUTINE ACT_DEL_ACE =
1336 1779 1
1337 1780 1 **
1338 1781 1
1339 1782 1 FUNCTIONAL DESCRIPTION:
1340 1783 1
1341 1784 1 This routine deletes the current ACE (bounded by the first and last
1342 1785 1 line segment pointers) and stores it for later retrieval. The ACE
1343 1786 1 is also deleted from the object's ACL is necessary.
1344 1787 1
1345 1788 1 CALLING SEQUENCE:
1346 1789 1 ACT_DEL_ACE ()
1347 1790 1
1348 1791 1 INPUT PARAMETERS:
1349 1792 1 none
1350 1793 1
1351 1794 1 IMPLICIT INPUTS:
1352 1795 1 OWN storage
1353 1796 1
1354 1797 1 OUTPUT PARAMETERS:
1355 1798 1 none
1356 1799 1
1357 1800 1 IMPLICIT OUTPUTS:
1358 1801 1 none
1359 1802 1
1360 1803 1 ROUTINE VALUE:
1361 1804 1 1 if successful
1362 1805 1 error status otherwise
1363 1806 1
1364 1807 1 SIDE EFFECTS:
1365 1808 1 The line segment table is updated as necessary, ACE line pointers
1366 1809 1 are updated, and the object's ACL is updated as necessary.
1367 1810 1
1368 1811 1 --
1369 1812 1
1370 1813 2 BEGIN
1371 1814 2
1372 1815 2 ! Check to see if the ACE is untouchable. If so, it cannot be deleted.
1373 1816 2
1374 1817 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1375 1818 2 THEN
1376 1819 2 BEGIN
1377 1820 2 SIGNAL (AED$ NOMODIFY);
1378 1821 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1379 1822 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1380 1823 2 TERM CHAR = 0;
1381 1824 2 RETURN 1;
1382 1825 2 END;
1383 1826 2
1384 1827 2 ! Delete anything currently in the delete ACE buffer.
1385 1828 2
1386 1829 2 UNTIL REMQUE (.AED_Q_DEL_ACE[LINE_L_FLINK], REMOVED_LINE)
1387 1830 2 DO
1388 1831 2 BEGIN
1389 1832 2 REMOVED_ACE = .REMOVED_LINE[LINE_L_BINACE];
1390 1833 2 IF .REMOVED_LINE[LINE_V_BEGINACE]
```


ACT_DEL_ACE - delete current ACE

```
1391      1834      AND .REMOVED_ACE NEQ 0
1392      1835      THEN DEALLOCATE (.REMOVED_ACE[ACESB_SIZE], REMOVED_ACE);
1393      1836      DEALLOCATE (.REMOVED_LINE[LINE_W_SIZE] + $BYTEOFFSET (LINE_T_TEXT),
1394      1837      REMOVED_LINE);
1395      1838      END;
1396      1839      NEW_TEXT_LINE = AED_REPSEGMENT ();
1397      1840      AED_POSITION (.AED_L_FIRSTLINE);
1398      1841      AED_L_BEGINLINE = .AED_L_BEGINLINE[LINE_L_BLINK];
1399      1842      TEMP_LINE = .AED_B_LINE = 1;
1400      1843      NEW_TEXT_LINE = .AED_L_FIRSTLINE;
1401      1844      DO
1402      1845      BEGIN
1403      1846      AED_L_FIRSTLINE = .AED_L_FIRSTLINE[LINE_L_FLINK];
1404      1847      REMOVE (NEW_TEXT_LINE[LINE_L_FLINK], REMOVED_LINE);
1405      1848      IF NOT .REMOVED [LINE_V_DUMMY]
1406      1849      THEN INSQUE (REMOVED_LINE[LINE_L_FLINK],
1407      1850      .AED_Q_DEL_ACE[LINE_L_BLINK]);
1408      1851      TEMP_LINE = .TEMP_LINE + 1;
1409      1852      NEW_TEXT_LINE = .AED_L_FIRSTLINE;
1410      1853      END
1411      1854      UNTIL .REMOVED_LINE EQL .AED_L_LASTLINE;
1412      1855      AED_L_STATUS = .AED_UPDATEACL (0);
1413      1856      AED_L_FLAGS[AED_V_INSERT] = AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
1414      1857      AED_L_FLAGS[AED_V_MODIFIED] = 0;
1415      1858      IF NOT .AED_L_STATUS
1416      1859      THEN
1417      1860      BEGIN
1418      1861      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1419      1862      RETURN 0;
1420      1863      END;
1421      1864      IF .AED_L_FIRSTLINE EQLA AED_Q_LINETABLE[LINE_L_FLINK]
1422      1865      THEN
1423      1866      BEGIN
1424      1867      BUFFER_INDEX = 0;
1425      1868      AED_L_FLAGS[AED_V_ENDACL] = 1;
1426      1869      AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
1427      1870      AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
1428      1871      INSQUE (AED_T_CURLINE[LINE_L_FLINK],
1429      1872      .AED_Q_LINETABLE[LINE_L_BLINK]);
1430      1873      AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
1431      1874      AED_L_FIRSTLINE[LINE_Q_FLAGS] = LINE_M_BEGINACE;
1432      1875      AED_L_CURACE = 0;
1433      1876      IF .AED_L_FLAGS[AED_V_PROMPT]
1434      1877      THEN
1435      1878      BEGIN
1436      1879      AED_B_ACETYPE = 0;
1437      1880      AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
1438      1881      AED_SELECTFIELD (BUFFER_INDEX);
1439      1882      END;
1440      1883      END
1441      1884      ELSE
1442      1885      BEGIN
1443      1886      BUFFER_INDEX = 0;
1444      1887      AED_COPYSEGMENT (.AED_L_FIRSTLINE);
1445      1888      INSQUE (AED_T_CURLINE[LINE_L_FLINK],
1446      1889      .AED_L_FIRSTLINE[LINE_L_BLINK]);
1447      1890      AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
```

ACT_DEL_ACE - delete current ACE

```
1448 1891 3 AED_W_TOTALSIZE = .AED_L_FIRSTLINE[LINE_W_SIZE];
1449 1892 3 UNTIL .AED_L_LASTLINE[LINE_V_ENDACE]
1450 1893 3 DO
1451 1894 4 BEGIN
1452 1895 4 IF .AED_L_LASTLINE EQA AED_T_CURLINE
1453 1896 4 THEN AED_C_LASTLINE = .AED_C_LASTLINE[LINE_L_FLINK];
1454 1897 4 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
1455 1898 4 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
1456 1899 4 END;
1457 1900 3 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
1458 1901 3 END;
1459 1902 3 AED_B_COLUMN = .BUFFER_INDEX + 1;
1460 1903 3 AED_L_BEGINLINE = .AED_L_BEGINLINE[LINE_L_FLINK];
1461 1904 3
1462 1905 3 ! Now repaint the display. This is done by either scrolling down and repainting
1463 1906 3 ! the first part of the display or repainting from the current position to the
1464 1907 3 ! end of the display (or the end of the ACL).
1465 1908 3
1466 1909 3 IF .AED_B_LINE LEQ 10
1467 1910 3 THEN
1468 1911 4 BEGIN
1469 1912 4 INCR J FROM 0 TO .TEMP_LINE - .AED_B_LINE
1470 1913 4 DO
1471 1914 5 BEGIN
1472 1915 5 IF .J EQ 0 THEN SCR$SET_CURSOR (20,1); ! **** TEMP ****
1473 1916 5 SCR$UP_SCROLL ();
1474 1917 5 END;
1475 1918 5 NEW TEXT LINE = .AED_L_BEGINLINE;
1476 1919 5 INCR J FROM 1 TO .AED_B_LINE
1477 1920 5 DO
1478 1921 6 BEGIN
1479 1922 6 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
1480 1923 6 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
1481 1924 6 SCR$SET_CURSOR (.J, 1);
1482 1925 6 AED_PUTOUTPUT (ECHO_DESC);
1483 1926 6 SCR$ERASE_LINE (.J, ECHO_DESC[DSC$W_LENGTH] + 1);
1484 1927 6 IF .NEW_TEXT_LINE[LINE_V_REPLACE] THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1485 1928 6 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1486 1929 6 END;
1487 1930 5 END
1488 1931 3 ELSE
1489 1932 4 BEGIN
1490 1933 4 IF .AED_L_FLAGS[AED_V_ENDACL]
1491 1934 4 THEN NEW_TEXT_LINE = AED_T_CURLINE
1492 1935 4 ELSE NEW_TEXT_LINE = .AED_T_CURLINE[LINE_L_FLINK];
1493 1936 4 INCR J FROM .AED_B_LINE TO 20
1494 1937 4 DO
1495 1938 5 BEGIN
1496 1939 5 IF .NEW_TEXT_LINE EQA AED_Q_LINETABLE[LINE_L_FLINK]
1497 1940 5 THEN
1498 1941 6 BEGIN
1499 1942 6 IF .J LSS 20 THEN SCR$ERASE_PAGE (.J, 1);
1500 1943 6 EXITLOOP;
1501 1944 6 END;
1502 1945 5 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
1503 1946 5 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
1504 1947 5 SCR$SET_CURSOR (.J, 1);
```

```
! End of routine ACT_DEL_ACE
```

				OFFC	00000	ACT_DEL_ACE:		
			5B	00000000G	00	9E	00002	.WORD
			5A	00000000G	00	9E	00009	MOVAB
			59	00000000G	8F	D0	00010	MOVAB
			58	00000000G	00	9E	00017	MOVAB
			57	00000000G	00	9E	0001E	MOVAB
			56	0000'	CF	9E	00025	MOVAB
			55	0000'	CF	9E	0002A	MOVAB
			5E		04	C2	0002F	SUBL2
			50		65	D0	00032	MOVL
47		0A	A0		04	E1	00035	BBC
OE		CO	A5		03	E1	0003A	BBC
					01	DD	0003F	PUSHL
					15	DD	00041	PUSHL
			68		02	FB	00043	CALLS
					01	DD	00046	PUSHL
					15	DD	00048	PUSHL
			67		02	FB	0004A	CALLS
					59	DD	0004D	PUSHL
					01	FB	0004F	CALLS
	0B	00000000G	A5		03	E1	00056	BBC
		CO	7E	E0	A5	9A	0005B	MOVZBL
			7E	E4	A5	9A	0005F	MOVZBL
			67		02	FB	00063	CALLS
				00000000*	8F	D5	00066	TSTL
					10	13	0006C	BEQL
00000000*	8F		03		00	ED	0006E	CMPZV
					04	18	00078	BGEQ
			D4	A5	59	D0	0007A	MOVL
					021A	31	0007E	BRW
			F8	A6	10	B5	0F	00081
					35	1D	00086	BVS
			50	F8	A6	D0	00088	MOVL
			FC	A6	0C	A0	D0	0008C
			10	0A	A0	E9	00091	BLBC
					0E	13	00095	BEQL
				FC	A6	9F	00097	PUSHAB
			04	AE	FC	B6	9A	0009A
								MOVZBL
								Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11
								SCR\$ERASE LINE, R11
								LIB\$FREE VM, R10
								#AED\$ NOMODIFY, R9
								SCR\$ERASE PAGE, R8
								SCR\$SET CURSOR, R7
								NEW TEXT LINE, R6
								AED_L FIRSTLINE, R5
								#4, -SP
								AED_L FIRSTLINE, R0
								#4, -10(R0), 4\$
								#3, AED_L_FLAGS, 1\$
								#1
								#21
								#2, SCR\$ERASE_PAGE
								#1
								#21
								#2, SCR\$SET_CURSOR
								R9
								#1, LIB\$SIGNAL
								#3, AED_L_FLAGS, 2\$
								AED_B_COLUMN, -(SP)

		04	AE	9F	0009F	PUSHAB	4(SP)		
	6A	02	FB	000A2	CALLS	#2, LIB\$FREE_VM			
		F8	A6	9F	000A5	PUSHAB	REMOVED_LINE	1837	
	50	F8	A6	D0	000A8	MOVL	REMOVED_LINE, R0		
04	AE	08	A0	3C	000AC	MOVZWL	8(R0), 4(SP)		
04	AE	14	C0	000B1	ADDL2	#20, 4(SP)			
	6A	04	AE	9F	000B5	PUSHAB	4(SP)		
		02	FB	000B8	CALLS	#2, LIB\$FREE_VM		1829	
		C4	11	000BB	BRB	4\$		1839	
0000G	CF	00	FB	000BD	CALLS	#0, AED_REPSEGMENT			
	66	50	D0	000C2	MOVL	R0, NEW_TEXT_LINE			
		65	DD	000C5	PUSHL	AED_L_FIRSTLINE		1840	
0000G	CF	01	FB	000C7	CALLS	#1, AED_POSITION			
	50	08	A5	D0	000CC	MOVL	AED_L_BEGINLINE, R0	1841	
		04	A0	D0	000D0	MOVL	4(R0), AED_L_BEGINLINE		
	F4	E4	A5	9A	000D5	MOVZBL	AED_B_LINE, TEMP_LINE	1842	
		F4	A6	D7	000DA	DECL	TEMP_LINE		
	66	65	D0	000DD	MOVL	AED_L_FIRSTLINE, NEW_TEXT_LINE		1843	
	51	65	D0	000E0	MOVL	AED_L_FIRSTLINE, R1		1846	
	65	61	D0	000E3	MOVL	(R1), AED_L_FIRSTLINE			
	F8	00	B6	0F	000E6	REMQUE	@NEW_TEXT_LINE, REMOVED_LINE	1847	
	50	F8	A6	D0	000EB	MOVL	REMOVED_LINE, R0	1848	
04	0A	02	E0	000EF	BBS	#2, 10(R0), 8\$			
	14	60	0E	000F4	INSQUE	(R0), @AED_Q_DEL_ACE+4		1850	
		F4	A6	D6	000F8	INCL	TEMP_LINE	1851	
	51	65	D0	000FB	MOVL	AED_L_FIRSTLINE, R1		1852	
	66	51	D0	000FE	MOVL	R1, NEW_TEXT_LINE			
	04	A5	F8	A6	D1	00101	CMPL	REMOVED_LINE, AED_L_LASTLINE	1854
			DB	12	00106	BNEQ	7\$		
			7E	D4	00108	CLRL	-(SP)	1855	
0000G	CF	01	FB	0010A	CALLS	#1, AED_UPDATEACL			
	4C	50	D0	0010F	MOVL	R0, AED_L_STATUS			
	C0	8F	AA	00113	BICW2	#24704, AED_L_FLAGS		1856	
		4C	A5	E8	00119	BLBS	AED_L_STATUS, 9\$	1858	
	C1	08	8A	0011D	BICB2	#8, AED_L_FLAGS+1		1861	
		0184	31	00121	BRW	29\$		1862	
		E8	A6	D4	00124	CLRL	BUFFER_INDEX	1867	
	50	F0	A5	9E	00127	MOVAB	AED_Q_LINETABLE, R0	1864	
	50	65	D1	0012B	CMPL	AED_L_FIRSTLINE, R0			
		3A	12	0012E	BNEQ	10\$			
	C0	8F	A8	00130	BISW2	#16416, AED_L_FLAGS		1869	
		78	A5	B4	00136	CLRW	SEGMENT_SIZE	1870	
		0284	C5	B4	00139	CLRW	AED_W_TOTALSIZE		
	F4	70	A5	0E	0013D	INSQUE	AED_T_CURLINE, @AED_Q_LINETABLE+4	1872	
	50	70	A5	9E	00142	MOVAB	AED_T_CURLINE, R0	1873	
	04	50	D0	00146	MOVL	R0, AED_L_LASTLINE			
		50	D0	0014A	MOVL	R0, AED_L_FIRSTLINE			
	0A	01	B0	0014D	MOVW	#1, 10(R0)		1874	
		F0	A5	D4	00151	CLRL	AED_L_CURACE	1875	
		C1	A5	95	00154	TSTB	AED_L_FLAGS+1	1876	
		60	18	00157	BGEQ	14\$			
		68	A5	94	00159	CLRB	AED_B_ACETYPE	1879	
	C2	08	8A	0015C	BICB2	#8, AED_L_FLAGS+2		1880	
		E8	A6	9F	00160	PUSHAB	BUFFER_INDEX	1881	
0000G	CF	01	FB	00163	CALLS	#1, AED_SELECTFIELD			
		4F	11	00168	BRB	14\$		1864	
		65	DD	0016A	PUSHL	AED_L_FIRSTLINE		1887	

	0000G	CF		01	FB	0016C	CALLS	#1, AED COPSEGMENT	
		50		65	DO	00171	MOVL	AED_L_FIRSTLINE, R0	1889
	04	B0	70	A5	OE	00174	INSQUE	AED_T_CURLINE, 34(R0)	
		50	70	A5	9E	00179	MOVAB	AED_T_CURLINE, R0	1890
	04	A5		50	DO	0017D	MOVL	R0, AED_L_LASTLINE	
		65		50	DO	00181	MOVL	R0, AED_L_FIRSTLINE	
		51		65	DO	00184	MOVL	AED_L_FIRSTLINE, R1	1891
	0284	C5	08	A1	B0	00187	MOVW	8(RT), AED_W_TOTALSIZE	
		50	04	A5	DO	0018D	MOVL	AED_L_LASTLINE, R0	1892
1E	0A	A0		01	EO	00191	BBS	#1, 10(R0), 13\$	
		52	70	A5	9E	00196	MOVAB	AED_T_CURLINE, R2	1895
		52		50	D1	0019A	CMPD	R0, R2	
				04	12	0019D	BNEQ	12\$	
	04	A5		60	DO	0019F	MOVL	(R0), AED_L_LASTLINE	1896
	04	A5	04	B5	DO	001A3	MOVL	AED_L_LASTLINE, AED_L_LASTLINE	1897
		50	04	A5	DO	001A8	MOVL	AED_L_LASTLINE, R0	1898
	0284	C5	08	A0	A0	001AC	ADDW2	8(R0), AED_W_TOTALSIZE	
				DD	11	001B2	BRB	11\$	1892
	FC	A5	0C	A1	DO	001B4	MOVL	12(R1), AED_L_CURACE	1900
EO	A5	E8		01	81	001B9	ADDB3	#1, BUFFER INDEX, AED_B_COLUMN	1902
		08		B5	DO	001BF	MOVL	AED_L_BEGINLINE, AED_L_BEGINLINE	1903
		52	E4	A5	9A	001C4	MOVZBL	AED_B_LINE, R2	1909
		0A		52	91	001C8	CMPB	R2, #0	
				65	1A	001CB	BGTRU	21\$	
54	F4	A6		52	C3	001CD	SUBL3	R2, TEMP_LINE, R4	1912
		53		01	CE	001D2	MNEGL	#1, J	
				10	11	001D5	BRB	17\$	
				07	12	001D7	BNEQ	16\$	1915
				01	DD	001D9	PUSHL	#1	
				14	DD	001DB	PUSHL	#20	
		67		02	FB	001DD	CALLS	#2, SCR\$SET_CURSOR	
	00000000G	00		00	FB	001E0	CALLS	#0, SCR\$UP_SCROLL	1916
EC		53		54	F3	001E7	AOBLEQ	R4, J, 15\$	1912
		66	08	A5	DO	001EB	MOVL	AED_L_BEGINLINE, NEW_TEXT_LINE	1918
		53	E4	A5	9A	001EF	MOVZBL	AED_B_LINE, R3	1919
				52	D4	001F3	CLRL	J	
				35	11	001F5	BRB	20\$	
		50		66	DO	001F7	MOVL	NEW TEXT LINE, R0	1922
	EC	A6	08	A0	B0	001FA	MOVW	8(R0), ECHO_DESC	
	FO	A6	14	A0	9E	001FF	MOVAB	20(R0), ECHO_DESC+4	1923
				01	DD	00204	PUSHL	#1	1924
				52	DD	00206	PUSHL	J	
		67		02	FB	002C8	CALLS	#2, SCR\$SET_CURSOR	
	0000G	CF	EC	A6	9F	0020B	PUSHAB	ECHO_DESC	1925
		7E	EC	01	FB	0020E	CALLS	#1, AED PUTOUTPUT	
				A6	3C	00213	MOVZWL	ECHO_DESC, -(SP)	1926
				6E	D6	00217	INCL	(SP)	
				52	DD	00219	PUSHL	J	
		6B		02	FB	0021B	CALLS	#2, SCR\$ERASE_LINE	
		50		66	DO	0021E	MOVL	NEW TEXT LINE, R0	1927
03	0A	A0		03	E1	00221	BBC	#3, 10(R0), 19\$	
		66		60	DO	00226	MOVL	(R0), NEW TEXT LINE	
		76		96	DO	00229	MOVL	AED_L_LASTLINE, NEW_TEXT_LINE	1928
C7		52		53	F3	0022F	AOBLEQ	R3, J, 18\$	1919
				58	11	00230	BRB	27\$	1909
06	C0	A5		05	E1	00232	BBC	#5, AED L FLAGS, 22\$	1933
		66	70	A5	9E	00237	MOVAB	AED_T_CURLINE, NEW_TEXT_LINE	1934

			04	11	0023B		BRB	23\$		
	66	70	A5	D0	0023D	22\$:	MOVL	AED_T_CURLINE, NEW_TEXT_LINE		1935
			52	D7	00241	23\$:	DECL	J		1936
			41	11	00243		BRB	26\$		
	50	FO	A5	9E	00245	24\$:	MOVAB	AED_Q LINETABLE, R0		1939
	50		66	D1	00249		CMPL	NEW_TEXT_LINE, R0		
			0E	12	0024C		BNEQ	25\$		
	14		52	D1	0024E		CMPL	J, #20		1942
			37	18	00251		BGEQ	27\$		
			01	DD	00253		PUSHL	#1		
			52	DD	00255		PUSHL	J		
	68		02	FB	00257		CALLS	#2, SCR\$ERASE_PAGE		
			2E	11	0025A		BRB	27\$		1941
	50		66	D0	0025C	25\$:	MOVL	NEW TEXT LINE, R0		1945
EC	A6	DB	A0	B0	0025F		MOVW	8(R0), ECHO_DESC		
FO	A6	14	A0	9E	00264		MOVAB	20(R0), ECHO_DESC+4		1946
			01	DD	00269		PUSHL	#1		1947
			52	DD	0026B		PUSHL	J		
	67		02	FB	0026D		CALLS	#2, SCR\$SET_CURSOR		
		EC	A6	9F	00270		PUSHAB	ECHO_DESC		1948
0000G	CF		01	FB	00273		CALLS	#1, AED PUTOUTPUT		
	7E	EC	A6	3C	00278		MOVZWL	ECHO_DESC, -(SP)		1949
			6E	D6	0027C		INCL	(SP)		
			52	DD	0027E		PUSHL	J		
	6B		02	FB	00280		CALLS	#2, SCR\$ERASE_LINE		
	76		96	D0	00283		MOVL	NEW_TEXT_LINE, NEW_TEXT_LINE		1950
BB	52		14	F3	00286	26\$:	AOBLEQ	#20, J, 24\$		1936
	A5		10	88	0028A	27\$:	BISB2	#16, AED_L_FLAGS+1		1953
C1	7E	E0	A5	9A	0028E		MOVZBL	AED_B_COLUMN, -(SP)		1954
	7E	E4	A5	9A	00292		MOVZBL	AED_B_LINE, -(SP)		
0000G	CF		02	FB	00296		CALLS	#2, AED SET_CURSOR		
C1	A5	2008	8F	AA	0029B	28\$:	BICW2	#8200, AED_C_FLAGS+1		1956
		10	A6	94	002A1		CLRB	TERM_CHAR		1957
	50		01	D0	002A4		MOVL	#1, R0		1958
				04	002A7		RET			
			50	D4	002AB	29\$:	CLRL	R0		1960
			04	002AA			RET			

; Routine Size: 683 bytes, Routine Base: \$CODE\$ + 0E38

ACT_UNDEL_CHR - insert deleted character

```
1519 1961 1 %SBTTL 'ACT UNDEL CHR - insert deleted character'
1520 1962 1 ROUTINE ACT_UNDEL_CHR =
1521 1963 1
1522 1964 1 !++
1523 1965 1
1524 1966 1 FUNCTIONAL DESCRIPTION:
1525 1967 1
1526 1968 1 This routine retrieves the previously deleted character and inserts
1527 1969 1 it into the line segment at the current cursor position. The
1528 1970 1 cursor position is unchanged.
1529 1971 1
1530 1972 1 CALLING SEQUENCE:
1531 1973 1 ACT_UNDEL_CHR ()
1532 1974 1
1533 1975 1 INPUT PARAMETERS:
1534 1976 1 none
1535 1977 1
1536 1978 1 IMPLICIT INPUTS:
1537 1979 1 OWN storage
1538 1980 1
1539 1981 1 OUTPUT PARAMETERS:
1540 1982 1 none
1541 1983 1
1542 1984 1 IMPLICIT OUTPUTS:
1543 1985 1 none
1544 1986 1
1545 1987 1 ROUTINE VALUE:
1546 1988 1 1 if successful
1547 1989 1 error status otherwise
1548 1990 1
1549 1991 1 SIDE EFFECTS:
1550 1992 1 The line segment table is updated as necessary, ACE line pointers
1551 1993 1 are updated, and the object's ACL is updated as necessary.
1552 1994 1
1553 1995 1 !--
1554 1996 1
1555 1997 2 BEGIN
1556 1998 2
1557 1999 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1558 2000 2 IF .AED_B_DEL_CHAR EQL 0
1559 2001 2 THEN
1560 2002 2 BEGIN
1561 2003 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1562 2004 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1563 2005 2 TERM_CHAR = 0;
1564 2006 2 RETURN 1;
1565 2007 2 END;
1566 2008 2
1567 2009 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
1568 2010 2
1569 2011 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1570 2012 2 THEN
1571 2013 2 BEGIN
1572 2014 2 SIGNAL (AED$ NOMODIFY);
1573 2015 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1574 2016 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1575 2017 2 TERM_CHAR = 0;
```

```
1576 2018 3 RETURN 1;  
1577 2019 3 END;  
1578 2020 3  
1579 2021 3 ! Retrieve the deleted character.  
1580 2022 3  
1581 2023 3 AED_L_FLAGS[AED_V_GOLDKEY] = 0;  
1582 2024 3 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;  
1583 2025 3 TERM_CHAR = .AED_B_DEL_CHAR;  
1584 2026 3 RETURN 1;  
1585 2027 3  
1586 2028 1 END;
```

! End of routine ACT_UNDEL_CHR

				001C 00000 ACT_UNDEL_CHR:				
			54 00000000G	8F D0 00002	WORD	Save R2,R3,R4	1962	
			53 00000000G	00 9E 00009	MOVL	#AED\$_NOMODIFY, R4		
			52 0000'	CF 9E 00010	MOVAB	SCR\$SET_CURSOR, R3		
	02	A2		20 8A 00015	MOVAB	AED_L_FLAGS, R2		
			68	A2 95 00019	BICB2	#32, AED_L_FLAGS+2	1999	
				4F 13 0001C	TSTB	AED_B_DEL_CHAR	2000	
			40	BEQL	3\$			
		50		A2 D0 0001E	MOVL	AED_L_FIRSTLINE, R0	2011	
52	0A	A0		04 E1 00022	BBC	#4, -10(R0), 4\$		
12		62		03 E1 00027	BBC	#3, AED_L_FLAGS, 1\$	2014	
				01 DD 0002B	PUSHL	#1		
				15 DD 0002D	PUSHL	#21		
		00000000G	00	02 FB 0002F	CALLS	#2, SCR\$ERASE_PAGE		
				01 DD 00036	PUSHL	#1		
				15 DD 00038	PUSHL	#21		
		63		02 FB 0003A	CALLS	#2, SCR\$SET_CURSOR		
				54 DD 0003D	PUSHL	R4		
		00000000G	00	01 FB 0003F	CALLS	#1, LIB\$SIGNAL		
0B				03 E1 00046	BBC	#3, AED_L_FLAGS, 2\$		
			20	A2 9A 0004A	MOVZBL	AED_B_COLUMN, -(SP)		
			24	A2 9A 0004E	MOVZBL	AED_B_LINE, -(SP)		
			63	02 FB 00052	CALLS	#2, SCR\$SET_CURSOR		
		00000000*		8F D5 00055	TSTL	#<AED\$_NOMODIFY&7>		
				10 13 0005B	BEQL	3\$		
00000000*	8F	14	A2	03	00 ED 0005D	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOMODIFY&7>	
				04 1B 00067	BGEQ	3\$		
		14	A2		54 D0 00069	MOVL	R4, AED_L_WORSTERR	
		01	A2	2008	8F AA 0006D	BICW2	#8200, AED_L_FLAGS+1	2016
				0000'	CF 94 00073	CLRB	TERM_CHAR	2017
					0C 11 00077	BRB	5\$	2018
		01	A2	2008	8F AA 00079	BICW2	#8200, AED_L_FLAGS+1	2024
		0000'	CF	68	A2 90 0007F	MOVAB	AED_B_DEL_CHAR, TERM_CHAR	2025
			50		01 D0 00085	MOVL	#1, R0	2026
					04 00088	RET		2028

; Routine Size: 137 bytes. Routine Base: \$CODE\$ + 10E3

ACT_UNDEL_WRD - insert deleted word

```
1588 2029 1 ZSBTTL 'ACT UNDEL WRD - insert deleted word'
1589 2030 1 ROUTINE ACT_UNDEL_WRD =
1590 2031 1
1591 2032 1 **
1592 2033 1
1593 2034 1 FUNCTIONAL DESCRIPTION:
1594 2035 1
1595 2036 1 This routine retrieves the previously deleted word and inserts it
1596 2037 1 into the current line segment starting at the current cursor position.
1597 2038 1 If a delete word was used previously, the cursor position is not
1598 2039 1 changed. If a rubout word was used, the cursor is moved to the end
1599 2040 1 in the retrieved word.
1600 2041 1
1601 2042 1 CALLING SEQUENCE:
1602 2043 1 ACT_UNDEL_WRD ( )
1603 2044 1
1604 2045 1 INPUT PARAMETERS:
1605 2046 1 none
1606 2047 1
1607 2048 1 IMPLICIT INPUTS:
1608 2049 1 OWN storage
1609 2050 1
1610 2051 1 OUTPUT PARAMETERS:
1611 2052 1 none
1612 2053 1
1613 2054 1 IMPLICIT OUTPUTS:
1614 2055 1 none
1615 2056 1
1616 2057 1 ROUTINE VALUE:
1617 2058 1 1 if successful
1618 2059 1 error status otherwise
1619 2060 1
1620 2061 1 SIDE EFFECTS:
1621 2062 1 The line segment table is updated as necessary, ACE line pointers
1622 2063 1 are updated, and the object's ACL is updated as necessary.
1623 2064 1
1624 2065 1 --
1625 2066 1
1626 2067 2 BEGIN
1627 2068 2
1628 2069 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
1629 2070 2
1630 2071 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1631 2072 2 THEN
1632 2073 2 BEGIN
1633 2074 2 SIGNAL (AED$ NOMODIFY);
1634 2075 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1635 2076 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1636 2077 2 TERM CHAR = 0;
1637 2078 2 RETURN 1;
1638 2079 2 END;
1639 2080 2
1640 2081 2 ! Retrieve the deleted word.
1641 2082 2
1642 2083 2 CH$COPY (.SEGMENT SIZE - .BUFFER INDEX,
1643 2084 2 INPUT_BUFFER[.BUFFER_INDEX],
1644 2085 2 0.
```

```
1645 2086 2 512 - .BUFFER_INDEX - .AED_Q_DEL_WORD[DSCSW_LENGTH],
1646 2087 INPUT_BUFFER[.BUFFER_INDEX +
1647 2088 .AED_Q_DEL_WORD[DSCSW_LENGTH]];
1648 2089 CHSMOVE (.AED_Q_DEL_WORD[DSCSW_LENGTH],
1649 2090 .AED_Q_DEL_WORD[DSCSA_POINTER],
1650 2091 INPUT_BUFFER[.BUFFER_INDEX]);
1651 2092 SEGMENT_SIZE = .SEGMENT_SIZE + .AED_Q_DEL_WORD[DSCSW_LENGTH];
1652 2093 ECHO_DESC[DSCSW_LENGTH] = .SEGMENT_SIZE - .BUFFER_INDEX;
1653 2094 ECHO_DESC[DSCSA_POINTER] = INPUT_BUFFER[.BUFFER_INDEX];
1654 2095 IF .AED_L_FLAGS[AED_V RUBWORD]
1655 2096 THEN BUFFER_INDEX = .BUFFER_INDEX + .AED_Q_DEL_WORD[DSCSW_LENGTH];
1656 2097 IF .SEGMENT_SIZE GEQ .AED_L_PAGEWIDTH
1657 2098 THEN AED_SEGSPLIT (BUFFER_INDEX, 0, 1, 0)
1658 2099 ELSE AED_PUTOUTPUT (ECHO_DESC);
1659 2100 AED_B_COLUMN = .BUFFER_INDEX + 1;
1660 2101 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1661 2102 IF .AED_Q_DEL_WORD[DSCSW_LENGTH] NEQ 0
1662 2103 THEN
1663 2104 BEGIN
1664 2105 AED_L_FLAGS[AED_V_MODIFIED] = 1;
1665 2106 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1666 2107 END;
1667 2108 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1668 2109 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1669 2110 TERM_CHAR = 0;
1670 2111 RETURN 1;
1671 2112
1672 2113 1 END;
```

! End of routine ACT_UNDEL_WRD

				OFFC 00000 ACT_UNDEL_WRD:								
				5B	00000000G	00	9E	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	2030	
				5A	0000'	CF	9E	000C9	MOVAB	SCR\$SET_CURSOR, R11		
				59	0000'	CF	9E	0000E	MOVAB	BUFFER_INDEX, R10		
				50	40	A9	DD	00013	MOVL	AED_L_FIRSTLINE, R0	2071	
51	0A			A0		04	E1	00017	BBC	#4, -10(R0), 48		
12				69		03	E1	0001C	BBC	#3, AED_L_FLAGS, 18	2074	
						01	DD	00020	PUSHL	#1		
						15	DD	00022	PUSHL	#21		
	00000000G	00				02	FB	00024	CALLS	#2, SCR\$ERASE_PAGE		
						01	DD	0002B	PUSHL	#1		
						15	DD	0002D	PUSHL	#21		
				6B		02	FB	0002F	CALLS	#2, SCR\$SET_CURSOR		
	00000000G	00			00000000G	8F	DD	00032	18:	PUSHL	#AED\$_NOMODIFY	
						01	FB	00038	CALLS	#1, LIB\$SIGNAL		
0B				69		03	E1	0003F	BBC	#3, AED_L_FLAGS, 28		
				7E	20	A9	9A	00043	MOVZBL	AED_B_COLUMN, -(SP)		
				7E	24	A9	9A	00047	MOVZBL	AED_B_LINE, -(SP)		
				6B		02	FB	0004B	CALLS	#2, SCR\$SET_CURSOR		
					00000000*	8F	DD	0004E	28:	TSTL	#<AED\$_NOMODIFY>	
						14	13	00054	BEQL	38		
00000000*	8F	14	A9	03		00	ED	00056	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOMODIFY>		
						0B	18	00060	BGEQ	38		

14	A9	00000000G	8F	D0	00062	MOVL	#AED\$_NOMODIFY, AED_L_WORSTERR	2075
			0087	31	0006A	BRW	8\$	2083
			56	6A	D0	0006D		
		00B8	C9	3C	00070	MOVL	BUFFER_INDEX, R6	
			52	56	C2	00075	MOVZWL	SEGMENT_SIZE, R2
			52	9E	00078	SUBL2	R6, R2	
		00C4	C946	9E	00078	MOVAB	INPUT_BUFFER[R6], R8	2084
			60	A9	3C	0007E	MOVZWL	AED_Q_DEL_WORD, R7
			57	9E	00082	MOVAB	-512(R7)[R6], R1	2086
		FE00	C746	CE	00088	MNEGL	R1, R1	
			51	C1	00088	ADDL3	R7, R6, R0	2088
			56	2C	0008F	MOVCS	R2, (R8), #0, R1, INPUT_BUFFER[R0]	
			68		00094			
		00C4	C940	28	00098	MOVCS	R7, @AED_Q_DEL_WORD+4, (R8)	2091
			57	A0	0009D	ADDW2	R7, SEGMENT_SIZE	2092
		64	B9	A3	000A2	SUBW3	R6, SEGMENT_SIZE, ECHO_DESC	2093
		00B8	C9	D0	000A9	MOVL	R8, ECHO_DESC+4	2094
			08	E1	000AD	BBC	#1, AED_C_FLAGS+1, 5\$	2095
		04	AA	C0	000B2	ADDL2	R7, BUFFER_INDEX	2096
			03	01	000B5	CMPZV	#0, #16, SEGMENT_SIZE, AED_L_PAGEWIDTH	2097
			01	19	000BD	BLSS	6\$	
			6A	7D	000BF	MOVQ	#1, -(SP)	2098
			10	7E	D4	000C2	CLRL	-(SP)
			7E	5A	DD	000C4	PUSHL	R10
		0000G	CF	04	FB	000C6	CALLS	#4, AED_SEGSPLIT
				08	11	000CB	BRB	7\$
				AA	9F	000CD	PUSHAB	ECHO_DESC
		04	AA	01	FB	000D0	CALLS	#1, AED_PUTOUTPUT
				01	81	000D5	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN
		0000G	CF	01	9A	000DA	MOVZBL	AED_B_COLUMN, -(SP)
			6A	9A	000DE	MOVZBL	AED_B_LINE, -(SP)	
			7E	02	FB	000E2	CALLS	#2, AED_SET_CURSOR
			7E	02	FB	000E2	TSTW	AED_Q_DEL_WORD
		0000G	CF	60	A9	B5	BEQL	8\$
				08	13	000EA	BISB2	#128, AED_L_FLAGS
				8F	88	000EC	BICB2	#16, AED_C_FLAGS+1
		01	A9	10	8A	000F0	BICW2	#8200, AED_L_FLAGS+1
				8F	AA	000F4	CLRB	TERM_CHAR
		01	A9	8F	AA	000FA	MOVL	#1, R0
				AA	94	000FA	RET	
			50	01	D0	000FD		
				04	00100			

; Routine Size: 257 bytes, Routine Base: \$CODE\$ + 116C

ACT_UNDEL_LIN - insert deleted line

```
1674 2114 1 %SBTTL 'ACT_UNDEL_LIN - insert deleted line'
1675 2115 1 ROUTINE ACT_UNDEL_LIN =
1676 2116 1
1677 2117 1 ++
1678 2118 1
1679 2119 1 FUNCTIONAL DESCRIPTION:
1680 2120 1
1681 2121 1 This routine retrieves the previously deleted line and inserts it
1682 2122 1 into the current line segment starting at the current cursor position.
1683 2123 1 If a delete line was used previously, the cursor position is not
1684 2124 1 changed. If a rubout line was used, the cursor is moved to the end
1685 2125 1 in the retrieved line.
1686 2126 1
1687 2127 1 CALLING SEQUENCE:
1688 2128 1 ACT_UNDEL_LIN ( )
1689 2129 1
1690 2130 1 INPUT PARAMETERS:
1691 2131 1 none
1692 2132 1
1693 2133 1 IMPLICIT INPUTS:
1694 2134 1 OWN storage
1695 2135 1
1696 2136 1 OUTPUT PARAMETERS:
1697 2137 1 none
1698 2138 1
1699 2139 1 IMPLICIT OUTPUTS:
1700 2140 1 none
1701 2141 1
1702 2142 1 ROUTINE VALUE:
1703 2143 1 1 if successful
1704 2144 1 error status otherwise
1705 2145 1
1706 2146 1 SIDE EFFECTS:
1707 2147 1 The line segment table is updated as necessary, ACE line pointers
1708 2148 1 are updated, and the object's ACL is updated as necessary.
1709 2149 1
1710 2150 1 --
1711 2151 1
1712 2152 2 BEGIN
1713 2153 2
1714 2154 2 ! If the current ACE is marked as untouchable, no modifications are allowed.
1715 2155 2
1716 2156 2 IF .AED_L_FIRSTLINE[LINE_V_NOTOUCH]
1717 2157 2 THEN
1718 2158 2 BEGIN
1719 2159 2 SIGNAL (AED$NOMODIFY);
1720 2160 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1721 2161 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1722 2162 2 TERM CHAR = 0;
1723 2163 2 RETURN 1;
1724 2164 2 END;
1725 2165 2
1726 2166 2 ! Retrieve the deleted line.
1727 2167 2
1728 2168 2 CH$COPY (.SEGMENT SIZE - .BUFFER INDEX,
1729 2169 2 INPUT_BUFFER[.BUFFER_INDEX],
1730 2170 2 0,
```



```
1731 2171 2 512 = .BUFFER_INDEX - .AED_Q_DEL_LINE[DSC$W_LENGTH],
1732 2172 2 INPUT_BUFFER[.BUFFER_INDEX + .AED_Q_DEL_LINE[DSC$W_LENGTH]];
1733 2173 2 CH$MOVE (.AED_Q_DEL_LINE[DSC$W_LENGTH],
1734 2174 2 .AED_Q_DEL_LINE[DSC$A_POINTER],
1735 2175 2 INPUT_BUFFER[.BUFFER_INDEX]);
1736 2176 2 SEGMENT_SIZE = .SEGMENT_SIZE + .AED_Q_DEL_LINE[DSC$W_LENGTH];
1737 2177 2 ECHO_DESC[DSC$W_LENGTH] = .SEGMENT_SIZE - .BUFFER_INDEX;
1738 2178 2 ECHO_DESC[DSC$A_POINTER] = INPUT_BUFFER[.BUFFER_INDEX];
1739 2179 2 IF .AED_L_FLAGS[AED_V_DELBOL]
1740 2180 2 THEN BUFFER_INDEX = .BUFFER_INDEX + .AED_Q_DEL_LINE[DSC$W_LENGTH];
1741 2181 2 IF .SEGMENT_SIZE GEQ .AED_L_PAGEWIDTH
1742 2182 2 THEN AED_SEGSPLIT (BUFFER_INDEX, 0, 1, 0)
1743 2183 2 ELSE AED_PUTOUTPUT (ECHO_DESC);
1744 2184 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
1745 2185 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1746 2186 2 IF .AED_Q_DEL_LINE[DSC$W_LENGTH] NEQ 0
1747 2187 2 THEN
1748 2188 2 BEGIN
1749 2189 2 AED_L_FLAGS[AED_V_MODIFIED] = 1;
1750 2190 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
1751 2191 2 END;
1752 2192 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1753 2193 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1754 2194 2 TERM_CHAR = 0;
1755 2195 2 RETURN 1;
1756 2196 2
1757 2197 1 END;
```

! End of routine ACT_UNDEL_LIN

OFFC 00000 ACT_UNDEL_LIN:										
									Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	2115
		5B	00000000G	00	9E	00002			WORD	
		5A	0000	CF	9E	00009			MOVAB	
		59	0000	CF	9E	0000E			MOVAB	
		50	40	A9	D0	00013			MOVL	
51	0A	A0		04	E1	00017			BBC	2156
12		69		03	E1	0001C			BBC	
				01	DD	00020			PUSHL	
				15	DD	00022			PUSHL	
		00000000G	00	02	FB	00024			CALLS	
				01	DD	0002B			PUSHL	
				15	DD	0002D			PUSHL	
		6B		02	FB	0002F			CALLS	
		00000000G	00	8F	DD	00032	1\$:		PUSHL	
				01	FB	00038			CALLS	
0B		69		03	E1	0003F			BBC	
		7E	20	A9	9A	00043			MOVZBL	
		7E	24	A9	9A	00047			MOVZBL	
		6B		02	FB	0004B			CALLS	
		00000000*		8F	D5	0004E	2\$:		TSTL	
				14	13	00054			BEQL	
00000000*	8F	14	A9	03	00	ED	00056		CMPZV	
				08	18	00060			BGEQ	
		14	A9	00000000G	8F	D0	00062		MOVL	
									#AED\$_NOMODIFY, AED_L_WORSTERR	

51	50	56	0087	31	0006A	3\$:	BRW	8\$	2160		
	00	52	6A	D0	0006D	4\$:	MOVL	BUFFER_INDEX, R6	2168		
	68	52	00B8	C9	3C	00070	MOVZWL	SEGMENT_SIZE, R2			
		52	56	C2	00075		SUBL2	R6, R2			
		58	00C4	C946	9E	00078	MOVAB	INPUT_BUFFER[R6], R8	2169		
		57	58	A9	3C	0007E	MOVZWL	AED_Q_DEL_LINE, R7	2171		
		51	FE00	C746	9E	00082	MOVAB	-512(R7)[R6], R1			
		51		51	CE	00088	MNEGL	R1, R1			
		56		57	C1	0008B	ADDL3	R7, R6, R0	2172		
		68		52	2C	0008F	MOVCS	R2, (R8), #0, R1, INPUT_BUFFER[R0]			
			00C4	C940		00094					
				57	28	00098	MOVCS	R7, AED_Q_DEL_LINE+4, (R8)	2175		
				57	A0	0009D	ADDW2	R7, SEGMENT_SIZE	2176		
				56	A3	000A2	SUBW3	R6, SEGMENT_SIZE, ECHO_DESC	2177		
				58	D0	000A9	MOVL	R8, ECHO_DESC+4	2178		
				02	E1	000AD	BBC	#2, AED [FLAGS+1, 5\$	2179		
				57	C0	000B2	ADDL2	R7, BUFFER_INDEX	2180		
18	A9	00B8	C9	10	00	ED	000B5	5\$:	2181		
				7E	0E	19	000BD	BLSS	6\$		
					01	7D	000BF	MOVQ	#1, -(SP)	2182	
					7E	D4	000C2	CLRL	-(SP)		
					5A	DD	000C4	PUSHL	R10		
			0000G	CF	04	FB	000C6	CALLS	#4, AED_SEGSPLIT		
					08	11	000CB	BRB	7\$		
					04	AA	9F	000CD	6\$:	2183	
					01	FB	000D0	PUSHAB	ECHO_DESC		
			0000G	CF	01	81	000D5	CALLS	#1, AED_PUTOUTPUT		
					01	81	000D5	7\$:		2184	
					20	A9	9A	000DA	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	2185
					24	A9	9A	000DE	MOVZBL	AED_B_COLUMN, -(SP)	
						02	FB	000E2	MOVZBL	AED_B_LINE, -(SP)	
			0000G	CF	58	A9	B5	000E7	CALLS	#2, AED_SET_CURSOR	
					08	13	000EA	TSTW	AED_Q_DEL_LINE	2186	
					80	8F	88	000EC	BEQL	8\$	
					01	A9	10	8A	BISB2	#128, AED_L_FLAGS	2189
			01	A9	2008	8F	AA	000F0	BICB2	#16, AED [FLAGS+1	2190
					28	AA	94	000FA	BICW2	#8200, AED_L_FLAGS+1	2193
					01	D0	000FD	CLRB	TERM_CHAR	2194	
					04	00100	MOVL	#1, R0		2195	
							RET			2197	

; Routine Size: 257 bytes. Routine Base: \$CODE\$ + 1260

```
1759 2198 1 %SBTTL 'ACT_UNDEL_ACE - insert deleted ACE'
1760 2199 1 ROUTINE ACT_UNDEL_ACE =
1761 2200 1
1762 2201 1 ++
1763 2202 1
1764 2203 1 FUNCTIONAL DESCRIPTION:
1765 2204 1
1766 2205 1 This routine retrieves the previously deleted ACE and inserts it
1767 2206 1 into the ACL before the first line of the current ACE.
1768 2207 1
1769 2208 1 CALLING SEQUENCE:
1770 2209 1 ACT_UNDEL_ACE ()
1771 2210 1
1772 2211 1 INPUT PARAMETERS:
1773 2212 1 none
1774 2213 1
1775 2214 1 IMPLICIT INPUTS:
1776 2215 1 OWN storage
1777 2216 1
1778 2217 1 OUTPUT PARAMETERS:
1779 2218 1 none
1780 2219 1
1781 2220 1 IMPLICIT OUTPUTS:
1782 2221 1 none
1783 2222 1
1784 2223 1 ROUTINE VALUE:
1785 2224 1 1 if successful
1786 2225 1 error status otherwise
1787 2226 1
1788 2227 1 SIDE EFFECTS:
1789 2228 1 The line segment table is updated as necessary, ACE line pointers
1790 2229 1 are updated, and the object's ACL is updated as necessary.
1791 2230 1
1792 2231 1 --
1793 2232 1
1794 2233 2 BEGIN
1795 2234 2
1796 2235 2 LOCAL
1797 2236 2 CURRENT_LINE : REF $BBLOCK; ! Address of current line segment
1798 2237 2
1799 2238 2 IF .AED_Q_DEL_ACE[LINE_L_FLINK] EQ LA AED_Q_DEL_ACE[LINE_L_FLINK]
1800 2239 2 THEN
1801 2240 2 BEGIN
1802 2241 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
1803 2242 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1804 2243 2 TERM_CHAR = 0;
1805 2244 2 RETURN 1;
1806 2245 2 END;
1807 2246 2 NEW_TEXT LINE = AED_REPSEGMENT ();
1808 2247 2 IF .AED_L_FLAGS[AED_V_MODIFIED]
1809 2248 2 OR .AED_L_FLAGS[AED_V_INSERT]
1810 2249 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
1811 2250 2 THEN
1812 2251 2 BEGIN
1813 2252 2 FINISH ACE ();
1814 2253 2 IF .AED_L_FLAGS[AED_V_PROMPT]
1815 2254 2 AND .AED_L_FLAGS[AED_V_FIRSTCHAR]
```

```
1816 2255 3 THEN
1817 2256 4 BEGIN
1818 2257 4 NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
1819 2258 4 AED_W_TOTALSIZE = 0;
1820 2259 3 END;
1821 2260 3 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
1822 2261 3 IF AED_W_TOTALSIZE EQL 0
1823 2262 3 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1824 2263 3 AED_COMPRESS ();
1825 2264 3 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
1826 2265 3 IF NOT AED_L_STATUS
1827 2266 3 THEN
1828 2267 4 BEGIN
1829 2268 4 AED_L_FLAGS[AED_V_ACERROR] = 1;
1830 2269 4 AED_POSITION (.AED_L_FIRSTLINE);
1831 2270 4 AED_COPSEGMENT (.AED_L_FIRSTLINE);
1832 2271 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
1833 2272 4 .AED_C_FIRSTLINE[LINE_L_BLINK]);
1834 2273 4 IF AED_L_LASTLINE EQL AED_L_FIRSTLINE
1835 2274 4 THEN AED_C_LASTLINE = AED_T_CURLINE;
1836 2275 4 IF AED_C_BEGINLINE EQL AED_L_FIRSTLINE
1837 2276 4 THEN AED_C_BEGINLINE = AED_T_CURLINE;
1838 2277 4 AED_L_FIRSTLINE = AED_T_CURLINE;
1839 2278 4 IF AED_L_FIRSTLINE NEQ AED_L_LASTLINE
1840 2279 4 AND AED_C_FLAGS[AED_V_ENDACL]
1841 2280 4 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
1842 2281 4 BUFFER_INDEX = 0;
1843 2282 4 AED_B_COLUMN = 1;
1844 2283 4 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1845 2284 4 AED_L_FLAGS[AED_V_GOODREY] = 0;
1846 2285 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1847 2286 4 TERM_CHAR = 0;
1848 2287 4 RETURN 1;
1849 2288 3 END;
1850 2289 3 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
1851 2290 3 END;
1852 2291 2 AED_W_TOTALSIZE = 0;
1853 2292 2 AED_L_LASTLINE = AED_L_FIRSTLINE[LINE_L_BLINK];
1854 2293 2 AED_L_FIRSTLINE = 0;
1855 2294 2 REMOVED_LINE = AED_Q_DEL_ACE[LINE_L_FLINK];
1856 2295 2 CURRENT_LINE = AED_Q_DEL_ACE[LINE_L_FLINK];
1857 2296 2 UNTIL .CURRENT_LINE EQL AED_Q_DEL_ACE[LINE_L_FLINK]
1858 2297 2 DO
1859 2298 3 BEGIN
1860 2299 3 AED_L_STATUS = ALLOCATE (.CURRENT_LINE[LINE_W_SIZE] + $BYTEOFFSET (LINE_T_TEXT),
1861 2300 3 NEW_TEXT_LINE);
1862 2301 3 IF NOT AED_L_STATUS
1863 2302 3 THEN
1864 2303 4 BEGIN
1865 2304 4 SIGNAL (.AED_L_STATUS);
1866 2305 4 RETURN AED_C_STATUS;
1867 2306 3 END;
1868 2307 3 CH$MOVE (.CURRENT_LINE[LINE_W_SIZE] + $BYTEOFFSET (LINE_T_TEXT),
1869 2308 3 .CURRENT_LINE, .NEW_TEXT_LINE);
1870 2309 3 INSQUE (NEW_TEXT [LINE_L_FLINK], AED_L_LASTLINE[LINE_L_FLINK]);
1871 2310 3 IF AED_L_FIRSTLINE EQL 0 THEN AED_L_FIRSTLINE = .NEW_TEXT_LINE;
1872 2311 3 AED_L_LASTLINE = .NEW_TEXT_LINE;
```



```
1873 2312 3 AED W TOTALSIZE = .AED W TOTALSIZE + .NEW TEXT_LINE[LINE_W_SIZE];
1874 2313 3 CURRENT_LINE = .CURRENT_LINE[LINE_L_FLINK];
1875 2314 3 END;
1876 2315 2 IF .AED W TOTALSIZE GTR 0 THEN AED L_FLAGS[AED V_FIRSTCHAR] = 0;
1877 2316 2 AED L_FIRSTLINE[LINE_W_FLAGS] = LINE_M_BEGINACE;
1878 2317 2 AED_POSITION (.AED L_FIRSTLINE);
1879 2318 2 AED_COPSEGMENT (.AED L_FIRSTLINE);
1880 2319 2 INSQUE (AED T_CURLINE[LINE_L_FLINK], .AED_L_FIRSTLINE[LINE_L_BLINK]);
1881 2320 2 IF .AED L_FLAGS[AED_V_ENDACL]
1882 2321 2 THEN AED_C_CURACE = 0;
1883 2322 2 ELSE AED_L_CURACE = .SBBLOCK [.AED L_LASTLINE[LINE_L_FLINK], LINE_L_BINACE];
1884 2323 2 IF .AED_C_BEGINLINE EQL .AED L_FIRSTLINE THEN AED_C_BEGINLINE = AED_T_CURLINE;
1885 2324 2 IF .AED_L_LASTLINE EQL .AED_C_FIRSTLINE THEN AED_C_LASTLINE = AED_T_CURLINE;
1886 2325 2 AED_L_FIRSTLINE = AED_T_CURLINE;
1887 2326 2 AED_L_FIRSTLINE[LINE_C_BINACE] = 0;
1888 2327 2 AED_L_FLAGS[AED_V_INSERT] = AED_L_FLAGS[AED_V_MODIFIED] = 1;
1889 2328 2
1890 2329 2 ! Determine where the last line of the newly added ACE falls.
1891 2330 2
1892 2331 2 TEMP_LINE = .AED_B_LINE;
1893 2332 2 NEW_TEXT_LINE = .AED_L_FIRSTLINE;
1894 2333 2 UNTIL .NEW_TEXT_LINE EQL .AED_L_LASTLINE
1895 2334 2 DO
1896 2335 2 BEGIN
1897 2336 2 TEMP_LINE = .TEMP_LINE + 1;
1898 2337 2 IF .NEW_TEXT_LINE[LINE_V_REPLACE] THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1899 2338 2 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1900 2339 2 END;
1901 2340 2
1902 2341 2 ! Now repaint the display. This is done by either scrolling down and repainting
1903 2342 2 ! the first part of the display or repainting from the current position to the
1904 2343 2 ! end of the display (or the end of the ACL).
1905 2344 2
1906 2345 2 IF .AED_B_LINE GTR 1
1907 2346 2 THEN
1908 2347 2 BEGIN
1909 2348 2 IF .TEMP_LINE LEQ 10
1910 2349 2 THEN
1911 2350 2 BEGIN
1912 2351 2 INCR J FROM 0 TO .TEMP_LINE - .AED_B_LINE
1913 2352 2 DO
1914 2353 2 BEGIN
1915 2354 2 IF .J EQL 0 THEN SCR$SET_CURSOR (1,1); ! **** TEMP ****
1916 2355 2 SCR$DOWN_SCROLL ();
1917 2356 2 END;
1918 2357 2 NEW_TEXT_LINE = .AED_L_BEGINLINE;
1919 2358 2 INCR J FROM 1 TO .TEMP_LINE
1920 2359 2 DO
1921 2360 2 BEGIN
1922 2361 2 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
1923 2362 2 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
1924 2363 2 SCR$SET_CURSOR (J, 1);
1925 2364 2 AED_PUTOUTPUT (ECHO_DESC);
1926 2365 2 SCR$ERASE_LINE (.J, .ECHO_DESC[DSC$W_LENGTH] + 1);
1927 2366 2 IF .NEW_TEXT_LINE[LINE_V_REPLACE] THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1928 2367 2 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1929 2368 2 END;
```

```
1930 2369 4      END
1931 2370 3      ELSE
1932 2371 4      BEGIN
1933 2372 4      NEW_TEXT_LINE = .AED_T_CURLINE[LINE_L_FLINK];
1934 2373 4      INCR J FROM .AED_B_LINE TO 20
1935 2374 4      DO
1936 2375 5      BEGIN
1937 2376 5      ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
1938 2377 5      ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
1939 2378 5      SCR$SET CURSOR (.J, 1);
1940 2379 5      AED_PUTOUTPUT (ECHO_DESC);
1941 2380 5      SCR$ERASE LINE (.J, .ECHO_DESC[DSC$W_LENGTH] + 1);
1942 2381 5      NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
1943 2382 5      IF .NEW_TEXT_LINE EQCA AED_Q_LINETABLE[LINE_L_FLINK] THEN EXITLOOP;
1944 2383 4      END;
1945 2384 3      END;
1946 2385 2      END;
1947 2386 2      BUFFER_INDEX = 0;
1948 2387 2      AED_B_COLUMN = 1;
1949 2388 2      AED_SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
1950 2389 2      AED_L_FLAGS[AED_V_GO[DREY]] = 0;
1951 2390 2      AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
1952 2391 2      TERM_CHAR = 0;
1953 2392 2      RETURN 1;
1954 2393 2
1955 2394 1      END;
```

! End of routine ACT_UNDEL_ACE

OFFC 00000 ACT_UNDEL_ACE:									
	5B	00000000G	00	9E	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	2199	
	5A	0000	CF	9E	00009	MOVAB	SCR\$SET CURSOR, R11		
	59	0000	CF	9E	0000E	MOVAB	NEW_TEXT_LINE, R10		
	5E		04	C2	00013	SUBL2	AED_L_FIRSTLINE, R9		
	50	10	A9	9E	00016	MOVAB	#4, -SP		
	50	10	A9	D1	0001A	CMPL	AED_Q_DEL_ACE, R0	2238	
			03	12	0001E	BNEQ	AED_Q_DEL_ACE, R0		
			02B9	31	00020	BRW	1\$		
	0000G	CF	00	FB	00023	CALLS	39\$		
	6A		50	D0	00028	MOVL	#0, AED_REPSEGMENT	2246	
		C0	A9	95	0002B	TSTB	R0, NEW_TEXT_LINE		
			0D	19	0002E	BLSS	AED_L_FLAGS	2247	
08	C1	A9	05	E0	00030	BBS	2\$		
03	C1	A9	06	E0	00035	BBS	#5, AED_L_FLAGS+1, 2\$	2248	
			008C	31	0003A	BRW	#6, AED_L_FLAGS+1, 2\$	2249	
	0000V	CF	00	FB	0003D	CALLS	9\$		
		C1	A9	95	00042	TSTB	#0, FINISH_ACE	2252	
			10	18	00045	BGEQ	AED_L_FLAGS+1	2253	
0B	C1	A9	04	E1	00047	BBC	3\$		
	50		6A	D0	0004C	MOVL	#4, AED_L_FLAGS+1, 3\$	2254	
	0A	A0	04	88	0004F	BISB2	NEW_TEXT_LINE, R0	2257	
		0284	C9	B4	00053	CLRW	#4, -10(R0)		
	C1	A9	40	8F	8A	BICB2	AED_W_TOTALSIZE	2258	
		0284	C9	B5	0005C	TSTW	#64, AED_L_FLAGS+1	2260	
							AED_W_TOTALSIZE	2261	

			03	12	00060	BNEQ	4\$				
			9A	DO	00062	MOVL	NEW TEXT LINE, NEW_TEXT_LINE				2262
			00	FB	00065	CALLS	#0, AED COMPRESS				2263
			C9	3C	0006A	MOVZWL	AED_W_TOTALSIZE, -(SP)				2264
			01	FB	0006F	CALLS	#1, AED_UPDATEACL				
			50	DO	00074	MOVL	R0, AED_L_STATUS				2265
			A9	F8	00078	BLBS	AED_L_STATUS, 8\$				2268
			8F	88	0007C	BISB2	#64, AED_L_FLAGS				2269
			69	DD	00081	PUSHL	AED_L_FIRSTLINE				
			01	FB	00083	CALLS	#1, AED_POSITION				2270
			69	DD	00088	PUSHL	AED_L_FIRSTLINE				
			01	FB	0008A	CALLS	#1, AED_COPSEGMENT				2272
			69	DO	0008F	MOVL	AED_L_FIRSTLINE, R0				
			A9	0E	00092	INSQUE	AED_T_CURLINE, 34(R0)				2273
			A9	D1	00097	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE				
			05	12	0009B	BNEQ	5\$				2274
			A9	9E	0009D	MOVAB	AED_T_CURLINE, AED_L_LASTLINE				2275
			A9	D1	000A2	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE				
			05	12	000A6	BNEQ	6\$				2276
			A9	9E	000AB	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE				2277
			A9	9E	000AD	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE				2278
			69	D1	000B1	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE				
			09	13	000B5	BEQL	7\$				2279
			05	E1	000B7	BBC	#5, AED_L_FLAGS, 7\$				2280
			20	8A	000BC	BICB2	#32, AED_L_FLAGS				2281
			0205	31	000C0	BRW	38\$				2289
			8F	AA	000C3	BICW2	#8320, AED_L_FLAGS				2291
			C9	B4	000C9	CLRW	AED_W_TOTALSIZE				2292
			69	DO	000CD	MOVL	AED_L_FIRSTLINE, R0				
			A0	DO	000D0	MOVL	4(R0), AED_L_LASTLINE				2293
			69	D4	000D5	CLRL	AED_L_FIRSTLINE				2294
			A9	DO	000D7	MOVL	AED_Q_DEL_ACE, REMOVED_LINE				2295
			A9	DO	000DC	MOVL	AED_Q_DEL_ACE, CURRENT_LINE				2296
			A9	9E	000E0	MOVAB	AED_Q_DEL_ACE, R0				
			56	D1	000E4	CMPL	CURRENT_LINE, R0				
			03	12	000E7	BNEQ	11\$				
			009E	31	000E9	BRW	19\$				2300
			5A	DD	000EC	PUSHL	R10				
			A6	3C	000EE	MOVZWL	8(CURRENT_LINE), R7				
			14	C0	000F2	ADDL2	#20, R7				
			57	DO	000F5	MOVL	R7, 4(SP)				
			AE	9F	000F9	PUSHAB	4(SP)				
			02	FB	000FC	CALLS	#2, LIB\$GET_VM				
			50	DO	00103	MOVL	R0, VM_STATUS				
			58	E9	00106	BLBC	VM_STATUS, 12\$				
			00	2C	00109	MOVCS	#0, (SP), #0, R7, NEW_TEXT_LINE				
			BA		0010E						
			58	DO	00110	MOVL	VM_STATUS, AED_L_STATUS				2301
			A9	E8	00114	BLBS	AED_L_STATUS, 17\$				2304
			03	E1	00118	BBC	#3, AED_L_FLAGS, 13\$				
			01	DD	0011D	PUSHL	#1				
			15	DD	0011F	PUSHL	#21				
			02	FB	00121	CALLS	#2, SCR\$ERASE_PAGE				
			01	DD	00128	PUSHL	#1				
			15	DD	0012A	PUSHL	#21				
			02	FB	0012C	CALLS	#2, SCR\$SET_CURSOR				
			A9	DD	0012F	PUSHL	AED_L_STATUS				

			00	01	FB	00132	CALLS	#1, LIBSSIGNAL	
	OB	00000000G	A9	03	E1	00139	BBC	#3, AED L FLAGS, 148	
		C0	7E	A9	9A	0013E	MOVZBL	AED_B_COLUMN, -(SP)	
			7E	A9	9A	00142	MOVZBL	AED_B_LINE, -(SP)	
			6B	02	FB	00146	CALLS	#2, SCRSET CURSOR	
			50	A9	D0	00149	MOVL	AED_L_STATUS, R0	
			07	50	93	0014D	BITB	R0, #7	
				01	12	00150	BNEQ	15\$	
					04	00152	RET		
				00	EF	00153	EXTZV	#0, #3, R0, R1	
				00	ED	00158	CMPZV	#0, #3, AED_L_WORSTERR, R1	
				01	19	0015E	BLSS	16\$	
					04	00160	RET		
			D4	50	D0	00161	MOVL	R0, AED_L_WORSTERR	
			A9		04	00165	RET		
				57	28	00166	MOV C3	R7, (CURRENT LINE), @NEW TEXT LINE	2305
				BA	0E	0016B	INSQUE	@NEW TEXT LINE, @AED_L_LASTLINE	2308
				69	D5	00170	TSTL	AED_C_FIRSTLINE	2309
				03	12	00172	BNEQ	18\$	2310
				6A	D0	00174	MOVL	NEW TEXT LINE, AED_L_FIRSTLINE	
				6A	D0	00177	MOVL	NEW TEXT LINE, R0	2311
				50	D0	0017A	MOVL	R0, AED_C_LASTLINE	
			04	A0	A0	0017E	ADDW2	8(R0), AED W TOTALSIZE	2312
			0284	C9	D0	00184	MOVL	(CURRENT LINE), CURRENT LINE	2313
				56	D0	00184	BRW	10\$	2296
				FF56	31	00187	TSTM	AED_W_TOTALSIZE	2315
				C9	B5	0018A	BEQL	20\$	
				04	13	0018E	BICB2	#16, AED L FLAGS+1	
				10	8A	00190	MOVL	AED_L_FIRSTLINE, R0	2316
				69	D0	00194	MOVW	#1, -10(R0)	
				01	B0	00197	PUSHL	R0	2317
				50	DD	0019B	CALLS	#1, AED POSITION	
				01	FB	0019D	PUSHL	AED_L_FIRSTLINE	2318
				69	DD	001A2	CALLS	#1, AED COPSEGMENT	
				01	FB	001A4	MOVL	AED_L_FIRSTLINE, R0	2319
				69	D0	001A9	INSQUE	AED_T_CURLINE, @4(R0)	
				A9	0E	001AC	BBC	#5, AED L FLAGS, 21\$	2320
				05	E1	001B1	CLRL	AED_L_CURACE	2321
				A9	D4	001B6	BRB	22\$	
				09	11	001B9	MOVL	@AED_L_LASTLINE, R0	2322
				B9	D0	001BB	MOVL	12(R0), AED L CURACE	
				A0	D0	001BF	CML	AED_L_BEGINLINE, AED_L_FIRSTLINE	2323
				A9	D1	001C4	BNEQ	23\$	
				05	12	001C8	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	
				A9	9E	001CA	CML	AED_L_LASTLINE, AED_C_FIRSTLINE	2324
				A9	D1	001CF	BNEQ	24\$	
				05	12	001D3	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	2325
				A9	9E	001DA	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2326
				69	D0	001DE	MOVL	AED_L_FIRSTLINE, R0	
				A0	D4	001E1	CLRL	12(R0)	
				8F	AB	001E4	BISW2	#8320, AED L FLAGS	2327
				A9	9A	001EA	MOVZBL	AED_B_LINE, R2	2331
				52	D0	001EE	MOVL	R2, TEMP LINE	
				50	D0	001F2	MOVL	R0, NEW TEXT LINE	2332
				6A	D0	001F5	MOVL	NEW TEXT LINE, R0	2333
				50	D1	001F8	CML	R0, AED_C_LASTLINE	
				10	13	001FC	BEQL	27\$	

03	0A	A0	F4	AA	D6	001FE	INCL	TEMP LINE	2336
		6A		03	E1	00201	BBC	#3, TO(R0), 26\$	2337
		7A		60	D0	00206	MOVL	(R0), NEW_TEXT_LINE	
				9A	D0	00209	MOVL	@NEW_TEXT_LINE, NEW_TEXT_LINE	2338
		01		E7	11	0020C	BRB	25\$	2333
				52	91	0020E	CMPB	R2, #1	2345
		0A	F4	6D	1B	00211	BLEQU	34\$	
				AA	D1	00213	CMPL	TEMP_LINE, #10	2348
54	F4	AA		69	14	00217	BGTR	35\$	
		53		52	C3	00219	SUBL3	R2, TEMP_LINE, R4	2351
				01	CE	0021E	MNEGL	#1, J	
				10	11	00221	BRB	30\$	
				07	12	00223	BNEQ	29\$	2354
				01	DD	00225	PUSHL	#1	
				01	DD	00227	PUSHL	#1	
		6B		02	FB	00229	CALLS	#2, SCR\$SET_CURSOR	
00000000G		00		00	FB	0022C	CALLS	#0, SCR\$DOWN_SCROLL	2355
EC		53		54	F3	00233	AOBLEQ	R4, J, 28\$	2351
		6A	08	A9	D0	00237	MOVL	AED_L BEGINLINE, NEW_TEXT_LINE	2357
		53	F4	AA	D0	0023B	MOVL	TEMP_LINE, R3	2358
				52	D4	0023F	CLRL	J	
				39	11	00241	BRB	33\$	
		50		6A	D0	00243	MOVL	NEW TEXT LINE, R0	2361
	EC	AA	08	A0	B0	00246	MOVW	8(R0), ECHO_DESC	
	FO	AA	14	A0	9E	0024B	MOVAB	20(R0), ECHO_DESC+4	2362
				01	DD	00250	PUSHL	#1	2363
		6B		52	DD	00252	PUSHL	J	
				02	FB	00254	CALLS	#2, SCR\$SET_CURSOR	
		CF	EC	AA	9F	00257	PUSHAB	ECHO_DESC	2364
0000G		7E	EC	01	FB	0025A	CALLS	#1, AED_PUTOUTPUT	
				AA	3C	0025F	MOVZWL	ECHO_DESC, -(SP)	2365
				6E	D6	00263	INCL	(SP)	
				52	DD	00265	PUSHL	J	
00000000G		00		02	FB	00267	CALLS	#2, SCR\$ERASE_LINE	
		50		6A	D0	0026E	MOVL	NEW_TEXT_LINE, R0	2366
03	0A	A0		03	E1	00271	BBC	#3, -10(R0), 32\$	
		6A		60	D0	00276	MOVL	(R0), NEW_TEXT_LINE	
		7A		9A	D0	00279	MOVL	@NEW_TEXT_LINE, NEW_TEXT_LINE	2367
C3		52		53	F3	0027C	AOBLEQ	R3, J, 31\$	2358
				46	11	00280	BRB	38\$	2348
		6A	70	A9	D0	00282	MOVL	AED_T CURLINE, NEW_TEXT_LINE	2372
		53		6A	D0	00286	MOVL	NEW_TEXT_LINE, R3	2376
				52	D7	00289	DECL	J	
				37	11	0028B	BRB	37\$	
	EC	AA	08	A3	B0	0028D	MOVW	8(R3), ECHO_DESC	
	FO	AA	14	A3	9E	00292	MOVAB	20(R3), ECHO_DESC+4	2377
				01	DD	00297	PUSHL	#1	2378
		6B		52	DD	00299	PUSHL	J	
				02	FB	0029B	CALLS	#2, SCR\$SET_CURSOR	
		CF	EC	AA	9F	0029E	PUSHAB	ECHO_DESC	2379
0000G		7E	EC	01	FB	002A1	CALLS	#1, AED_PUTOUTPUT	
				AA	3C	002A6	MOVZWL	ECHO_DESC, -(SP)	2380
				6E	D6	002AA	INCL	(SP)	
				52	DD	002AC	PUSHL	J	
00000000G		00		02	FB	002AE	CALLS	#2, SCR\$ERASE_LINE	
		7A		9A	D0	002B5	MOVL	@NEW_TEXT_LINE, NEW_TEXT_LINE	2381
		53		6A	D0	002B8	MOVL	NEW_TEXT_LINE, R3	2382

	50	F0	A9	9E	002BB	MOVAB	AED_Q LINETABLE, R0	
	50		53	D1	002BF	CMPL	R3, -R0	
			04	13	002C2	BEQL	38\$	
C5	52		14	F3	002C4	AOBLEQ	#20, J, 36\$	2373
		E8	AA	D4	002C8	CLRL	BUFFER INDEX	2386
	E0		01	90	002CB	MOVB	#1, AED_B COLUMN	2387
	7E	E0	A9	9A	002CF	MOVZBL	AED_B COLUMN, -(SP)	2388
	7E	E4	A9	9A	002D3	MOVZBL	AED_B LINE, -(SP)	
0000G	CF		02	FB	002D7	CALLS	#2, AED SET CURSOR	
C1	A9	2008	8F	AA	002DC	BICW2	#8200, AED_C_FLAGS+1	2390
		10	AA	94	002E2	CLRB	TERM CHAR	2391
	50		01	D0	002E5	MOVL	#1, R0	2392
			04	002E8	RET			2394

; Routine Size: 745 bytes, Routine Base: \$CODE\$ + 136E

ACT_MOVE_WRD - move to word boundary

```
1957 2395 1 %SBTTL 'ACT_MOVE_WRD - move to word boundary'
1958 2396 1 ROUTINE ACT_MOVE_WRD =
1959 2397 1
1960 2398 1
1961 2399 1
1962 2400 1
1963 2401 1
1964 2402 1
1965 2403 1
1966 2404 1
1967 2405 1
1968 2406 1
1969 2407 1
1970 2408 1
1971 2409 1
1972 2410 1
1973 2411 1
1974 2412 1
1975 2413 1
1976 2414 1
1977 2415 1
1978 2416 1
1979 2417 1
1980 2418 1
1981 2419 1
1982 2420 1
1983 2421 1
1984 2422 1
1985 2423 1
1986 2424 1
1987 2425 1
1988 2426 1
1989 2427 1
1990 2428 1
1991 2429 1
1992 2430 2
1993 2431 2
1994 2432 2
1995 2433 2
1996 2434 3
1997 2435 3
1998 2436 3
1999 2437 4
2000 2438 4
2001 2439 4
2002 2440 4
2003 2441 4
2004 2442 3
2005 2443 4
2006 2444 4
2007 2445 3
2008 2446 4
2009 2447 4
2010 2448 4
2011 2449 4
2012 2450 5
2013 2451 5
```

ROUTINE ACT_MOVE_WRD =

++

FUNCTIONAL DESCRIPTION:

This routine goes to the next word boundary (first non-alphanumeric character) in either the forward or backward direction.

CALLING SEQUENCE:

ACT_MOVE_WRD ()

INPUT PARAMETERS:

none

IMPLICIT INPUTS:

OWN storage

OUTPUT PARAMETERS:

none

IMPLICIT OUTPUTS:

none

ROUTINE VALUE:

1 if successful
error status otherwise

SIDE EFFECTS:

The line segment table is updated as necessary, ACE line pointers are updated, and the object's ACL is updated as necessary.

--

BEGIN

IF NOT .AED_L_FLAGS[AED_V_BACKWARD]
THEN
BEGIN
IF .BUFFER_INDEX GEQ .SEGMENT_SIZE
THEN
BEGIN
AED_L_FLAGS[AED_V_GOLDKEY] = 0;
AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
TERM_CHAR = 0;
RETURN 1;
END;
WHILE (.BUFFER_CHAR GEQ 'A' AND .BUFFER_CHAR LEQ 'Z')
OR (.BUFFER_CHAR GEQ '0' AND .BUFFER_CHAR LEQ '9')
DO
BEGIN
BUFFER_INDEX = .BUFFER_INDEX + 1;
IF .BUFFER_INDEX GEQ .SEGMENT_SIZE
THEN
BEGIN
BUFFER_INDEX = .BUFFER_INDEX - 1;

ACT_MOVE_WRD - move to word boundary

```
2014 2452 5 EXITLOOP;
2015 2453 4 END;
2016 2454 3 END;
2017 2455 2 END
2018 2456 1 ELSE
2019 2457 BEGIN
2020 2458 BUFFER_INDEX = .BUFFER_INDEX - 2;
2021 2459 IF .BUFFER_INDEX GEQ 0
2022 2460 THEN
2023 2461 BEGIN
2024 2462 WHILE (.BUFFER_CHAR GEQ 'A' AND .BUFFER_CHAR LEQ 'Z')
2025 2463 OR (.BUFFER_CHAR GEQ '0' AND .BUFFER_CHAR LEQ '9')
2026 2464 DO
2027 2465 BEGIN
2028 2466 BUFFER_INDEX = .BUFFER_INDEX - 1;
2029 2467 IF .BUFFER_INDEX LSS 0 THEN EXITLOOP;
2030 2468 END;
2031 2469 END
2032 2470 ELSE BUFFER_INDEX = -1;
2033 2471 END;
2034 2472 BUFFER_INDEX = .BUFFER_INDEX + 1; ! First char of word
2035 2473 AED_B_COLUMN = .BUFFER_INDEX + 1;
2036 2474 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2037 2475 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2038 2476 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2039 2477 TERM_CHAR = 0;
2040 2478 RETURN 1;
2041 2479
2042 2480 1 END; ! End of routine ACT_MOVE_WRD
```

				0004 00000 ACT_MOVE_WRD:			
			52	0000' CF 9E 00002	MOVAB	Save R2	2396
			38	0000' CF E8 00007	BLBS	BUFFER_INDEX, R2	
62	0000' CF		10	00 00 0000C	CMPZV	AED_L_FLAGS+1, 4\$	2432
				76 15 00013	BLEQ	#0, #T6, SEGMENT_SIZE, BUFFER_INDEX	2435
			50	0000' CF 9E 00015 1\$:	MOVAB	10\$	
			50	00 B240 9A 0001A	MOVZBL	INPUT BUFFER, R0	2443
		41	8F	50 91 0001F	CMPB	@BUFFER_INDEX[R0], R0	
				06 1F 00023	BLSSU	R0, #65	
		5A	8F	50 91 00025	CMPB	2\$	
				0A 1B 00029	BLEQU	R0, #90	
			30	50 91 0002B 2\$:	CMPB	3\$	2444
				44 1F 0002E	BLSSU	R0, #48	
			39	50 91 00030	CMPB	9\$	
				3F 1A 00033	BGTRU	R0, #57	
				62 D6 00035 3\$:	INCL	9\$	
62	0000' CF		10	00 00 00037	CMPZV	BUFFER_INDEX	2447
				D5 14 0003E	BGTR	#0, #16, SEGMENT_SIZE, BUFFER_INDEX	2448
				62 D7 00040	DECL	1\$	
				30 11 00042	BRB	BUFFER_INDEX	2451
						9\$	2450
		62		02 C2 00044 4\$:	SUBL	#2, BUFFER_INDEX	2458
				28 19 00047	BLSS	8\$	2459

	51		62	D0	00049		MOVL	BUFFER_INDEX, R1		2462
	50	0000'CF	41	9A	0004C	5\$:	MOVZBL	INPUT_BUFFER[R1], R0		
41	8F		50	91	00052		CMPB	R0, #85		
			06	1F	00056		BLSSU	6\$		
5A	8F		50	91	00058		CMPB	R0, #90		
			0A	1B	0005C		BLEQU	7\$		
	30		50	91	0005E	6\$:	CMPB	R0, #48		2463
			11	1F	00061		BLSSU	9\$		
	39		50	91	00063		CMPB	R0, #57		
			0C	1A	00066		BGTRU	9\$		
			62	D7	00068	7\$:	DECL	BUFFER_INDEX		2466
	51		62	D0	0006A		MOVL	BUFFER_INDEX, R1		2467
			DD	18	0006D		BGEQ	5\$		
			03	11	0006F		BRB	9\$		
	62		01	CE	00071	8\$:	MNEGL	#1, BUFFER_INDEX		2470
			62	D6	00074	9\$:	INCL	BUFFER_INDEX		2472
0000' CF	62		01	81	00076		ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN		2473
	7E	0000'	CF	9A	0007C		MOVZBL	AED_B_COLUMN, -(SP)		2474
	7E	0000'	CF	9A	00081		MOVZBL	AED_B_LINE, -(SP)		
0000G	CF		02	FB	00086		CALLS	#2, AED_SET_CURSOR		
0000'	CF	2008	8F	AA	0008B	10\$:	BICW2	#8200, AED_C_FLAGS+1		2476
		28	A2	94	00092		CLRB	TERM_CHAR		2477
	50		01	D0	00095		MOVL	#1, R0		2478
			04	00	00098		RET			2480

; Routine Size: 153 bytes, Routine Base: \$CODE\$ + 1657

ACT_MOVE_ACE - move to ACE boundary

```
2044 2481 1 %SBTTL 'ACT_MOVE_ACE - move to ACE boundary'
2045 2482 1 ROUTINE ACT_MOVE_ACE =
2046 2483 1
2047 2484 1 **
2048 2485 1
2049 2486 1 FUNCTIONAL DESCRIPTION:
2050 2487 1
2051 2488 1     This routine advances or backs up over an ACE depending on the
2052 2489 1     state of the BACKWARD flag.
2053 2490 1
2054 2491 1 CALLING SEQUENCE:
2055 2492 1     ACT_MOVE_ACE ()
2056 2493 1
2057 2494 1 INPUT PARAMETERS:
2058 2495 1     none
2059 2496 1
2060 2497 1 IMPLICIT INPUTS:
2061 2498 1     OWN storage
2062 2499 1
2063 2500 1 OUTPUT PARAMETERS:
2064 2501 1     none
2065 2502 1
2066 2503 1 IMPLICIT OUTPUTS:
2067 2504 1     none
2068 2505 1
2069 2506 1 ROUTINE VALUE:
2070 2507 1     1 if successful
2071 2508 1     error status otherwise
2072 2509 1
2073 2510 1 SIDE EFFECTS:
2074 2511 1     The line segment table is updated as necessary, ACE line pointers
2075 2512 1     are updated, and the object's ACL is updated as necessary.
2076 2513 1
2077 2514 1 --
2078 2515 1
2079 2516 2 BEGIN
2080 2517 2
2081 2518 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
2082 2519 2 IF .AED_L_FLAGS[AED_V_MODIFIED]
2083 2520 2 OR .AED_L_FLAGS[AED_V_INSERT]
2084 2521 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
2085 2522 2 THEN
2086 2523 2     BEGIN
2087 2524 2         FINISH_ACE ();
2088 2525 2         IF .AED_L_FLAGS[AED_V_PROMPT]
2089 2526 2         AND .AED_L_FLAGS[AED_V_FIRSTCHAR]
2090 2527 2         THEN
2091 2528 2             BEGIN
2092 2529 2                 NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
2093 2530 2                 AED_W_TOTALSIZE = 0;
2094 2531 2             END;
2095 2532 2             AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2096 2533 2             IF .AED_W_TOTALSIZE EQL 0 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
2097 2534 2             AED_COMPRESS ();
2098 2535 2             AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
2099 2536 2             IF NOT .AED_L_STATUS
2100 2537 2             THEN
```

```
2101 2538 4 BEGIN
2102 2539 4 AED_L_FLAGS[AED_V_ACERROR] = 1;
2103 2540 4 AED_POSITION (.AED_L_FIRSTLINE);
2104 2541 4 AED_COPSEGMENT (.AED_L_FIRSTLINE);
2105 2542 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2106 2543 4 .AED_C_FIRSTLINE[LINE_L_BLINK]);
2107 2544 4 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
2108 2545 4 THEN AED_C_LASTLINE = AED_T_CURLINE;
2109 2546 4 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
2110 2547 4 THEN AED_C_BEGINLINE = AED_T_CURLINE;
2111 2548 4 AED_L_FIRSTLINE = AED_T_CURLINE;
2112 2549 4 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
2113 2550 4 AND .AED_C_FLAGS[AED_V_ENDACL]
2114 2551 4 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
2115 2552 4 BUFFER_INDEX = 0;
2116 2553 4 AED_B_COLUMN = 1;
2117 2554 4 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2118 2555 4 AED_L_FLAGS[AED_V_GODKEY] = 0;
2119 2556 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2120 2557 4 TERM_CHAR = 0;
2121 2558 4 RETURN 1;
2122 2559 4 END;
2123 2560 4 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
2124 2561 4 END;
2125 2562 4
2126 2563 4 IF NOT .AED_L_FLAGS[AED_V_BACKWARD]
2127 2564 4 THEN
2128 2565 4 BEGIN
2129 2566 4 AED_L_FIRSTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
2130 2567 4 IF .AED_L_FIRSTLINE EQA .AED_Q_LINETAB[LINE_L_FLINK]
2131 2568 4 THEN AED_C_FLAGS[AED_V_ENDACL] = 1;
2132 2569 4 AED_POSITION (.AED_L_FIRSTLINE);
2133 2570 4
2134 2571 4 ! If at the end of the ACL, set up to append. Otherwise setup the next line.
2135 2572 4
2136 2573 4 IF .AED_L_FLAGS[AED_V_ENDACL]
2137 2574 4 THEN
2138 2575 4 BEGIN
2139 2576 4 BUFFER_INDEX = 0;
2140 2577 4 AED_B_COLUMN = 1;
2141 2578 4 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
2142 2579 4 AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
2143 2580 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2144 2581 4 .AED_C_FIRSTLINE[LINE_L_BLINK]);
2145 2582 4 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
2146 2583 4 AED_L_FIRSTLINE[LINE_Q_FLAGS] = LINE_M_BEGINACE;
2147 2584 4 AED_L_CURACE = 0;
2148 2585 4 IF .AED_L_FLAGS[AED_V_PROMPT]
2149 2586 4 THEN
2150 2587 4 BEGIN
2151 2588 4 AED_B_ACETYPE = 0;
2152 2589 4 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
2153 2590 4 AED_SELECTFIELD (BUFFER_INDEX);
2154 2591 4 ECHO_DESC(DSC$W_LENGTH) = .AED_T_CURLINE[LINE_W_SIZE];
2155 2592 4 ECHO_DESC(DSC$A_POINTER) = AED_T_CURLINE[LINE_T_TEXT];
2156 2593 4 SCR$SET_CURSOR (.AED_B_LINE, 1);
2157 2594 4 AED_PUTOUTPUT (ECHO_DESC);
```

```
2158 2595 5      SCR$ERASE LINE (.AED B LINE, .SEGMENT_SIZE + 1);
2159 2596 5      AED_B_COLUMN = .BUFFER_INDEX + 1;
2160 2597 4      END;
2161 2598 4      END
2162 2599 3      ELSE
2163 2600 4      BEGIN
2164 2601 4      AED COPSEGMENT (.AED L FIRSTLINE);
2165 2602 4      INSQUE (AED T CURLINE[CINE L FLINK],
2166 2603 4      .AED [ FIRSTLINE[LINE L BLINK]]);
2167 2604 4      AED_L_FIRSTLINE = AED L LASTLINE = AED T CURLINE;
2168 2605 4      AED_W_TOTALSIZE = .AED [ FIRSTLINE[LINE_W_SIZE];
2169 2606 4      UNTIL .AED_L_LASTLINE[CINE_V_ENDACE]
2170 2607 4      DO
2171 2608 4      BEGIN
2172 2609 4      IF .AED L LASTLINE EQLA AED T CURLINE
2173 2610 4      THEN AED [ LASTLINE = .AED [ LASTLINE[LINE L FLINK];
2174 2611 4      AED_L_LASTLINE = .AED L LASTLINE[LINE L FLINK];
2175 2612 5      AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
2176 2613 4      END;
2177 2614 4      AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
2178 2615 4      BUFFER_INDEX = 0;
2179 2616 4      AED_B_COLUMN = 1;
2180 2617 4      END;
2181 2618 3      END
2182 2619 3      ELSE
2183 2620 4      BEGIN
2184 2621 4      AED_L_LASTLINE = AED L FIRSTLINE = .AED_L_FIRSTLINE[LINE_L_BLINK];
2185 2622 4      AED_W_TOTALSIZE = .AED_L_FIRSTLINE[LINE_W_SIZE];
2186 2623 4      UNTIL .AED_L_FIRSTLINE[LINE_V_BEGINACE]
2187 2624 4      DO
2188 2625 4      BEGIN
2189 2626 4      IF .AED L BEGINLINE EQLA .AED L FIRSTLINE
2190 2627 4      THEN AED [ BEGINLINE = .AED L_FIRSTLINE[LINE L BLINK];
2191 2628 4      AED_L_FIRSTLINE = .AED_L_FIRSTLINE[LINE L BLINK];
2192 2629 4      AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_FIRSTLINE[LINE_W_SIZE];
2193 2630 4      END;
2194 2631 4      AED_POSITION (.AED L FIRSTLINE);
2195 2632 4      AED COPSEGMENT (.AED L FIRSTLINE);
2196 2633 4      INSQUE (AED T CURLINE[CINE L FLINK],
2197 2634 4      .AED [ FIRSTLINE[LINE L BLINK]]);
2198 2635 4      IF .AED_L_BEGINLINE EQL .AED [ FIRSTLINE THEN AED L BEGINLINE = AED T CURLINE;
2199 2636 4      IF .AED_L_LASTLINE EQL .AED [ FIRSTLINE THEN AED [ LASTLINE = AED T CURLINE;
2200 2637 4      AED_L_FIRSTLINE = AED T CURLINE;
2201 2638 4      AED_W_TOTALSIZE = .AED [ FIRSTLINE[LINE_W_SIZE];
2202 2639 4      AED_L_CURACE = .AED L_FIRSTLINE[LINE_L_BINACE];
2203 2640 4      AED_L_FLAGS[AED_V_ENDACL] = 0;
2204 2641 4      AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2205 2642 4      BUFFER_INDEX = 0;
2206 2643 4      AED_B_COLUMN = 1;
2207 2644 4      END;
2208 2645 4      AED SET CURSOR (.AED B LINE, .BUFFER_INDEX + 1);
2209 2646 4      AED_L_FLAGS[AED_V_GOLDREY] = 0;
2210 2647 4      AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2211 2648 4      TERM CHAR = 0;
2212 2649 4      RETURN 1;
2213 2650 4
2214 2651 1      END;
```

! End of routine ACT_MOVE_ACE

				007C 00000 ACT_MOVE_ACE:			
08 03	0000G 18	56	0000G	CF 9E 00002	WORD	Save R2,R3,R4,R5,R6	2482
		55	0000G	CF 9E 00007	MOVAB	AED_COPSEGMENT, R6	
		54	0000'	CF 9E 0000C	MOVAB	AED_POSITION, R5	
		53	0000'	CF 9E 00011	MOVAB	BUFFER INDEX, R4	
		CF		00 FB 00016	MOVAB	AED_L_FIRSTLINE, R3	
		A4		50 D0 0001B	CALLS	#0, AED_REPSEGMENT	2518
			C0	A3 95 0001F	MOVL	R0, NEW_TEXT_LINE	
				0D 19 00022	TSTB	AED_L_FLAGS	2519
				05 E0 00024	BLSS	1\$	
				06 E0 00029	BBS	#5, AED_L_FLAGS+1, 1\$	2520
0C	0000V	C1 A3		0095 31 0002E	BBS	#6, AED_L_FLAGS+1, 1\$	2521
		C1 A3		00 FB 00031	BRW	8\$	
			C1	A3 95 00036	CALLS	#0, FINISH_ACE	2524
				11 18 00039	TSTB	AED_L_FLAGS+1	2525
		C1 A3		04 E1 0003B	BGEQ	2\$	
		50	18	A4 D0 00040	BBC	#4, AED_L_FLAGS+1, 2\$	2526
		0A A0		04 88 00044	MOVL	NEW_TEXT_LINE, R0	2529
				0284 C3 B4 00048	BISB2	#4, -10(R0)	
		C1 A3		40 BF 8A 0004C	CLRW	AED_W_TOTALSIZE	2530
				0284 C3 B5 00051	BICB2	#64, AED_L_FLAGS+1	2532
D4	0000G	18 A4		05 12 00055	TSTW	AED_W_TOTALSIZE	2533
		CF	18	B4 D0 00057	BNEQ	3\$	
		7E		00 FB 0005C	MOVL	#NEW TEXT LINE, NEW_TEXT_LINE	
		CF	0284	00 3C 00061	CALLS	#0, AED_COMPRESS	2534
		A3		01 FB 00066	MOVZWL	AED_W_TOTALSIZE, -(SP)	2535
		4C		50 D0 0006B	CALLS	#1, AED_UPDATEACL	
		4D	4C	A3 E8 0006F	MOVL	R0, AED_L_STATUS	
		C0 A3	40	8F 88 00073	BLBS	AED_L_STATUS, 7\$	2536
				63 DD 00078	BISB2	#64, AED_L_FLAGS	2539
		65		01 FB 0007A	PUSHL	AED_L_FIRSTLINE	2540
	04			63 DD 0007D	CALLS	#1, AED_POSITION	
		66		01 FB 0007F	PUSHL	AED_L_FIRSTLINE	2541
		50		63 D0 00082	CALLS	#1, AED_COPSEGMENT	
		B0	70	A3 0E 00085	MOVL	AED_L_FIRSTLINE, R0	2543
		63	04	A3 D1 0008A	INSQUE	AED_T_CURLINE, #4(R0)	
				05 12 0008E	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	2544
		A3	70	A3 9E 00090	BNEQ	4\$	
		63	08	A3 D1 00095	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	2545
				05 12 00099	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	2546
		A3	70	A3 9E 0009B	BNEQ	5\$	
	04	A3		A3 9E 000A0	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	2547
		63		63 D1 000A4	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2548
		A3		09 13 000A8	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	2549
				05 E1 000AA	BEQL	6\$	
		C0 A3		20 8A 000AF	BBC	#5, AED_L_FLAGS, 6\$	2550
		C0 A3		64 D4 000B3	BICB2	#32, AED_L_FLAGS	2551
		E0 A3		01 90 000B5	CLRL	BUFFER INDEX	2552
		7E	E0	A3 9A 000B9	MOVB	#1, AED_B_COLUMN	2553
				0171 31 000BD	MOVZBL	AED_B_COLUMN, -(SP)	2554
		C0 A3	2080	8F AA 000C0	BRW	24\$	
					BICW2	#8320, AED_L_FLAGS	2560

	03	C1	A3	E9	000C6	8\$:	BLBC	AED_L_FLAGS+1, 9\$	2563
			00E2	31	000CA		BRW	16\$	
	63	04	B3	D0	000CD	9\$:	MOVL	AED_L_LASTLINE, AED_L_FIRSTLINE	2566
	50	FO	A3	9E	000D1		MOVAB	AED_B_CINETABLE, R0	2567
	50		63	D1	000D5		CMPL	AED_L_FIRSTLINE, R0	
			04	12	000D8		BNEQ	10\$	
	C0	A3	20	88	000DA		BISB2	#32, AED_L_FLAGS	2568
			63	DD	000DE	10\$:	PUSHL	AED_L_FIRSTLINE	2569
	65		01	FB	000E0		CALLS	#1, AED_POSITION	
78	C0	A3	05	E1	000E3		BBC	#5, AED_L_FLAGS, 12\$	2573
			64	D4	000E8		CLRL	BUFFER_INDEX	2576
	E0	A3	01	90	000EA		MOVAB	#1, AED_B_COLUMN	2577
	C1	A3	40	8F	88	000EE	BISB2	#64, AED_C_FLAGS+1	2578
			78	A3	B4	000F3	CLRW	SEGMENT_SIZE	2579
		0284	C3	B4	000F6		CLRW	AED_W_TOTALSIZE	
	50		63	D0	000FA		MOVL	AED_L_FIRSTLINE, R0	2581
	04	B0	70	A3	0E	000FD	INSQUE	AED_T_CURLINE, 34(R0)	
	50		70	A3	9E	00102	MOVAB	AED_T_CURLINE, R0	2582
	04	A3	50	D0	00106		MOVL	R0, AED_L_LASTLINE	
	63		50	D0	0010A		MOVL	R0, AED_L_FIRSTLINE	
	0A	A0	01	B0	0010D		MOVW	#1, 10(R0)	2583
			FC	A3	D4	00111	CLRL	AED_L_CURACE	2584
			C1	A3	95	00114	TSTB	AED_L_FLAGS+1	2585
			44	18	00117		BGEQ	11\$	
		68	A3	94	00119		CLRB	AED_B_ACETYPE	2588
	C2	A3	08	8A	0011C		BICB2	#8, AED_L_FLAGS+2	2589
			54	DD	00120		PUSHL	R4	2590
	0000G	CF	01	FB	00122		CALLS	#1, AED_SELECTFIELD	
	04	A4	78	A3	B0	00127	MOVW	AED_T_CURLINE+8, ECHO_DESC	2591
	08	A4	0084	C3	9E	0012C	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	2592
			01	DD	00132		PUSHL	#1	2593
	7E		E4	A3	9A	00134	MOVZBL	AED_B_LINE, -(SP)	
	00000000G	00	02	FB	00138		CALLS	#2, SCRSET_CURSOR	
			04	A4	9F	0013F	PUSHAB	ECHO_DESC	2594
	0000G	CF	01	FB	00142		CALLS	#1, AED_PUTOUTPUT	
	7E		78	A3	3C	00147	MOVZWL	SEGMENT_SIZE, -(SP)	2595
			6E	D6	0014B		INCL	(SP)	
	7E		E4	A3	9A	0014D	MOVZBL	AED_B_LINE, -(SP)	
	00000000G	00	02	FB	00151		CALLS	#2, SCRERASE_LINE	
E0	A3	64	01	81	00158		ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	2596
			00CD	31	0015D	11\$:	BRW	23\$	2573
			63	DD	00160	12\$:	PUSHL	AED_L_FIRSTLINE	2601
	66		01	FB	00162		CALLS	#1, AED_COPSEGMENT	
	50		63	D0	00165		MOVL	AED_L_FIRSTLINE, R0	2603
	04	B0	70	A3	0E	00168	INSQUE	AED_T_CURLINE, 34(R0)	
	50		70	A3	9E	0016D	MOVAB	AED_T_CURLINE, R0	2604
	04	A3	50	D0	00171		MOVL	R0, AED_L_LASTLINE	
	63		50	D0	00175		MOVL	R0, AED_L_FIRSTLINE	
	52		63	D0	00178		MOVL	AED_L_FIRSTLINE, R2	2605
	0284	C3	08	A2	B0	0017B	MOVW	8(R2), AED_W_TOTALSIZE	
	51		04	A3	D0	00181	MOVL	AED_L_LASTLINE, R1	2606
1E	0A	A1	01	E0	00185	13\$:	BBS	#1, 10(R1), 15\$	
	50		70	A3	9E	0018A	MOVAB	AED_T_CURLINE, R0	2609
	50		51	D1	0018E		CMPL	R1, R0	
			04	12	00191		BNEQ	14\$	
	04	A3	61	D0	00193		MOVL	(R1), AED_L_LASTLINE	2610
	04	A3	04	B3	D0	00197	MOVL	AED_L_LASTLINE, AED_L_LASTLINE	2611

	51	04	A3	D0	0019C	MOVL	AED_L_LASTLINE, R1	2612	
0284	C3	08	A1	A0	001A0	ADDW2	8(RT), AED_W_TOTALSIZE	2613	
			DD	11	001A6	BRB	13\$	2606	
FC	A3	0C	A2	D0	001A8	15\$:	MOVL	12(R2), AED_L_CURACE	2614
			78	11	001AD	BRB	22\$	2615	
	50		63	D0	001AF	16\$:	MOVL	AED_L_FIRSTLINE, R0	2621
	50	04	A0	D0	001B2	MOVL	4(R0), R0		
	63		50	D0	001B6	MOVL	R0, AED_L_FIRSTLINE		
04	A3		50	D0	001B9	MOVL	R0, AED_L_LASTLINE		
	50		63	D0	001BD	MOVL	AED_L_FIRSTLINE, R0	2622	
0284	C3	08	A0	B0	001C0	MOVW	8(R0), AED_W_TOTALSIZE		
	50		63	D0	001C6	MOVL	AED_L_FIRSTLINE, R0	2623	
	1A	0A	A0	E8	001C9	17\$:	BLBS	10(R0), 19\$	
	50	08	A3	D1	001CD	CMPL	AED_L_BEGINLINE, R0	2626	
			05	12	001D1	BNEQ	18\$		
08	A3	04	A0	D0	001D3	MOVL	4(R0), AED_L_BEGINLINE	2627	
	63	04	A0	D0	001D8	18\$:	MOVL	4(R0), AED_L_FIRSTLINE	2628
	50		63	D0	001DC	MOVL	AED_L_FIRSTLINE, R0	2629	
0284	C3	08	A0	A0	001DF	ADDW2	8(R0), AED_W_TOTALSIZE		
			E2	11	001E5	BRB	17\$	2623	
	65		63	DD	001E7	19\$:	PUSHL	AED_L_FIRSTLINE	2631
			01	FB	001E9	CALLS	#1, AED_POSITION		
			63	DD	001EC	PUSHL	AED_L_FIRSTLINE	2632	
	66		01	FB	001EE	CALLS	#1, AED_COPSEGMENT		
	50		63	D0	001F1	MOVL	AED_L_FIRSTLINE, R0	2634	
04	B0	70	A3	0E	001F4	INSQUE	AED_T_CURLINE, 34(R0)		
	63	08	A3	D1	001F9	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	2635	
			05	12	001FD	BNEQ	20\$		
08	A3	70	A3	9E	001FF	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE		
	63	04	A3	D1	00204	20\$:	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	2636
			05	12	00208	BNEQ	21\$		
04	A3	70	A3	9E	0020A	MOVAB	AED_T_CURLINE, AED_L_LASTLINE		
	63	70	A3	9E	0020F	21\$:	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2637
	50		63	D0	00213	MOVL	AED_L_FIRSTLINE, R0	2638	
0284	C3	08	A0	B0	00216	MOVW	8(R0), AED_W_TOTALSIZE		
FC	A3	0C	A0	D0	0021C	MOVL	12(R0), AED_L_CURACE	2639	
CO	A3	4020	8F	AA	00221	BICW2	#16416, AED_L_FLAGS	2641	
			64	D4	0C227	22\$:	CLRL	BUFFER_INDEX	2642
	E0		01	90	00229	MOVB	#1, AED_B_COLUMN	2643	
7E	64		01	C1	0022D	23\$:	ADDL3	#1, BUFFER_INDEX, -(SP)	2645
	7E	E4	A3	9A	00231	24\$:	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	00235	CALLS	#2, AED_SET_CURSOR		
C1	A3	2008	8F	AA	0023A	BICW2	#8200, AED_L_FLAGS+1	2647	
		28	A4	94	00240	CLRB	TERM_CHAR	2648	
	50		01	D0	00243	MOVL	#1, R0	2649	
			04	00246	RET			2651	

; Routine Size: 583 bytes, Routine Base: \$CODE\$ + 16F0

ACT_MOVE_BOL - move to beginning of line

```
2216 2652 1 XSBTTL 'ACT MOVE BOL - move to beginning of line'
2217 2653 1 ROUTINE ACT_MOVE_BOL =
2218 2654 1
2219 2655 1 ++
2220 2656 1
2221 2657 1 FUNCTIONAL DESCRIPTION:
2222 2658 1
2223 2659 1 This routine positions the cursor to the beginning of the current
2224 2660 1 line segment.
2225 2661 1
2226 2662 1 CALLING SEQUENCE:
2227 2663 1 ACT_MOVE_BOL ()
2228 2664 1
2229 2665 1 INPUT PARAMETERS:
2230 2666 1 none
2231 2667 1
2232 2668 1 IMPLICIT INPUTS:
2233 2669 1 OWN storage
2234 2670 1
2235 2671 1 OUTPUT PARAMETERS:
2236 2672 1 none
2237 2673 1
2238 2674 1 IMPLICIT OUTPUTS:
2239 2675 1 none
2240 2676 1
2241 2677 1 ROUTINE VALUE:
2242 2678 1 1 if successful
2243 2679 1 error status otherwise
2244 2680 1
2245 2681 1 SIDE EFFECTS:
2246 2682 1 The line segment table is updated as necessary, ACE line pointers
2247 2683 1 are updated, and the object's ACL is updated as necessary.
2248 2684 1
2249 2685 1 --
2250 2686 1
2251 2687 2 BEGIN
2252 2688 2
2253 2689 2 BUFFER INDEX = 0;
2254 2690 2 AED_W_ITEMEND = 0;
2255 2691 2 AED_B_COLUMN = 1;
2256 2692 2 AED_SET_CURSOR (AED_B_LINE, 1);
2257 2693 2 AED_L_FLAGS[AED_V_GO[DREY]] = 0;
2258 2694 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2259 2695 2 TERM_CHAR = 0;
2260 2696 2 RETURN 1;
2261 2697 2
2262 2698 1 END;
```

! End of routine ACT_MOVE_BOL

```
0000 00000 ACT_MOVE_BOL:
0000' CF D4 00002 .WORD Save nothing
0000' CF B4 00006 CLRL BUFFER INDEX
0000' CF 01 90 0000A CLRW AED_W_ITEMEND
0000' CF 01 90 0000A MOVB #1, AED_B_COLUMN
```

```
: 2653
: 2689
: 2690
: 2691
```


ACT_MOVE_BOL - move to beginning of line

D 16
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDIT.SRC]AEDMAIN.B32;1

Page 83
(17)

0000G	7E	0000'	01	DD	0000F
0000'	CF		CF	9A	00011
	CF	2008	02	FB	00016
		0000'	8F	AA	00018
			CF	94	00022
	50		01	DO	00026
				04	00029

```

PUSHL    #1
MOVZBL   AED-B LINE, -(SP)
CALLS    #2, AED SET CURSOR
BICW2    #B200, AED-C_FLAGS+1
CLRB     TERM CHAR
MOVL     #1, RO
RET

```

2692
2694
2695
2696
2698

; Routine Size: 42 bytes, Routine Base: SCODES + 1937

ACT_MOVE_EOL - move to end of line

```
2264 2699 1 $SBTTL 'ACT MOVE EOL - move to end of line'
2265 2700 1 ROUTINE ACT_MOVE_EOL =
2266 2701 1
2267 2702 1 ++
2268 2703 1
2269 2704 1 FUNCTIONAL DESCRIPTION:
2270 2705 1
2271 2706 1 This routine positions the cursor to the end of the current line
2272 2707 1 segment.
2273 2708 1
2274 2709 1 CALLING SEQUENCE:
2275 2710 1 ACT_MOVE_EOL ()
2276 2711 1
2277 2712 1 INPUT PARAMETERS:
2278 2713 1 none
2279 2714 1
2280 2715 1 IMPLICIT INPUTS:
2281 2716 1 OWN storage
2282 2717 1
2283 2718 1 OUTPUT PARAMETERS:
2284 2719 1 none
2285 2720 1
2286 2721 1 IMPLICIT OUTPUTS:
2287 2722 1 none
2288 2723 1
2289 2724 1 ROUTINE VALUE:
2290 2725 1 1 if successful
2291 2726 1 error status otherwise
2292 2727 1
2293 2728 1 SIDE EFFECTS:
2294 2729 1 The line segment table is updated as necessary, ACE line pointers
2295 2730 1 are updated, and the object's ACL is updated as necessary.
2296 2731 1
2297 2732 1 --
2298 2733 1
2299 2734 2 BEGIN
2300 2735 2
2301 2736 2 BUFFER_INDEX = .SEGMENT_SIZE;
2302 2737 2 AED_W_ITEMEND = .BUFFER_INDEX;
2303 2738 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
2304 2739 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2305 2740 2 AED_L_FLAGS[AED_V_GOLDREY] = 0;
2306 2741 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2307 2742 2 TERM_CHAR = 0;
2308 2743 2 RETURN 1;
2309 2744 2
2310 2745 1 END;
```

! End of routine ACT_MOVE_EOL

```
000C 00000 ACT_MOVE_EOL:
53 0000' CF 9E 00002 .WORD Save R2,R3
52 0000' CF 9E 00007 MOVAB BUFFER_INDEX, R3
63 0098 C2 3C 0000C MOVAB AED_B_COLUMN, R2
MOVZWL SEGMENT_SIZE, BUFFER_INDEX
```

: 2700
:
: 2736

AEDSMAIN
V04-000

ACT_MOVE_EOL - move to end of line

F 16
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 85
(18)

62	0084	C2	63	B0	00011
		63	01	81	00016
		7E	62	9A	0001A
		7E	A2	9A	0001D
	0000G	CF	02	FB	00021
	E1	A2	8F	AA	00026
			A3	94	0002C
		50	01	D0	0002F
			04	00	00032

MOVW	BUFFER INDEX, AED_W_ITEMEND
ADDB3	#1, BUFFER INDEX, AED_B_COLUMN
MOVZBL	AED_B_COLUMN, -(SP)
MOVZBL	AED_B_LINE, -(SP)
CALLS	#2, AED SET CURSOR
BICW2	#8200, AED_L_FLAGS+1
CLRB	TERM CHAR
MOVL	#1, R0
RET	

: 2737
: 2738
: 2739
:
: 2741
: 2742
: 2743
: 2745

: Routine Size: 51 bytes, Routine Base: \$CODE\$ + 196,

ACT_UP - move up to previous line

```
2312 2746 1 XSBTTL 'ACT_UP - move up to previous line'
2313 2747 1 ROUTINE ACT_UP =
2314 2748 1
2315 2749 1 ++
2316 2750 1
2317 2751 1 FUNCTIONAL DESCRIPTION:
2318 2752 1
2319 2753 1 This routine moves the cursor up to the previous line segment. If
2320 2754 1 the current ACE has been modified but not entered, it is done before
2321 2755 1 moving the cursor. The cursor position in the new line segment
2322 2756 1 is minimized with the current cursor position and the end of the
2323 2757 1 line segment.
2324 2758 1
2325 2759 1 CALLING SEQUENCE:
2326 2760 1 ACT_UP ()
2327 2761 1
2328 2762 1 INPUT PARAMETERS:
2329 2763 1 none
2330 2764 1
2331 2765 1 IMPLICIT INPUTS:
2332 2766 1 OWN storage
2333 2767 1
2334 2768 1 OUTPUT PARAMETERS:
2335 2769 1 none
2336 2770 1
2337 2771 1 IMPLICIT OUTPUTS:
2338 2772 1 none
2339 2773 1
2340 2774 1 ROUTINE VALUE:
2341 2775 1 1 if successful
2342 2776 1 error status otherwise
2343 2777 1
2344 2778 1 SIDE EFFECTS:
2345 2779 1 The line segment table is updated as necessary, ACE line pointers
2346 2780 1 are updated, and the object's ACL is updated as necessary.
2347 2781 1
2348 2782 1 --
2349 2783 1
2350 2784 2 BEGIN
2351 2785 2
2352 2786 2 LOCAL
2353 2787 2 UP_LINE_SEGMENT : REF $BBLOCK; ! Address of previous line
2354 2788 2
2355 2789 2 AED_L_FLAGS[AED_V_INSERTTEXT] = 0; ! No item selection
2356 2790 2
2357 2791 2 IF .AED_T_CURLINE[LINE_L_BLINK] NEQA AED_Q_LINETABLE[LINE_L_FLINK]
2358 2792 2 THEN
2359 2793 2 BEGIN
2360 2794 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
2361 2795 2 IF .NEW_TEXT_LINE[LINE_V_BEGINACE]
2362 2796 2 THEN
2363 2797 2 BEGIN
2364 2798 2 IF .AED_L_FLAGS[AED_V_MODIFIED]
2365 2799 2 OR .AED_L_FLAGS[AED_V_INSERT]
2366 2800 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
2367 2801 2 THEN
2368 2802 2 BEGIN
```



```
2369 2803 5 FINISH ACE ();
2370 2804 5 IF .AED_L_FLAGS[AED_V_PROMPT]
2371 2805 5 AND .AED_C_FLAGS[AED_V_FIRSTCHAR]
2372 2806 5 THEN
2373 2807 5 BEGIN
2374 2808 5 NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
2375 2809 5 AED_W_TOTALSIZE = 0;
2376 2810 5 END;
2377 2811 5 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2378 2812 5 IF .AED_W_TOTALSIZE EQL 0
2379 2813 5 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
2380 2814 5 AED_COMPRESS ();
2381 2815 5 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
2382 2816 5 IF NOT .AED_L_STATUS
2383 2817 5 THEN
2384 2818 5 BEGIN
2385 2819 5 AED_L_FLAGS[AED_V_ACERROR] = 1;
2386 2820 5 AED_POSITION (.AED_L_FIRSTLINE);
2387 2821 5 AED_COPSEGMENT (.AED_L_FIRSTLINE);
2388 2822 5 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2389 2823 5 .AED_C_FIRSTLINE[LINE_L_BLINK]);
2390 2824 5 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
2391 2825 5 THEN AED_C_LASTLINE = AED_T_CURLINE;
2392 2826 5 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
2393 2827 5 THEN AED_C_BEGINLINE = AED_T_CURLINE;
2394 2828 5 AED_L_FIRSTLINE = AED_T_CURLINE;
2395 2829 5 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
2396 2830 5 AND .AED_C_FLAGS[AED_V_ENDACL]
2397 2831 5 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
2398 2832 5 BUFFER_INDEX = 0;
2399 2833 5 AED_B_COLUMN = 1;
2400 2834 5 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2401 2835 5 AED_L_FLAGS[AED_V_GODREY] = 0;
2402 2836 5 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2403 2837 5 TERM_CHAR = 0;
2404 2838 5 RETURN 1;
2405 2839 5 END;
2406 2840 5 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
2407 2841 5 END;
2408 2842 5 END;
2409 2843 5 UP_LINE_SEGMENT = .NEW_TEXT_LINE[LINE_L_BLINK];
2410 2844 5 AED_POSITION (.UP_LINE_SEGMENT);
2411 2845 5 AED_COPSEGMENT (.UP_LINE_SEGMENT);
2412 2846 5 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .UP_LINE_SEGMENT[LINE_L_BLINK]);
2413 2847 5 IF .AED_L_BEGINLINE EQL .UP_LINE_SEGMENT
2414 2848 5 THEN AED_C_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
2415 2849 5 IF .AED_C_FIRSTLINE EQL .UP_LINE_SEGMENT
2416 2850 5 THEN AED_C_FIRSTLINE = AED_T_CURLINE[LINE_L_FLINK];
2417 2851 5 IF .AED_T_CURLINE[LINE_V_ENDACE]
2418 2852 5 THEN
2419 2853 5 BEGIN
2420 2854 5 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
2421 2855 5 AED_W_TOTALSIZE = .AED_C_LASTLINE[LINE_W_SIZE];
2422 2856 5 UNTIL .AED_L_FIRSTLINE[LINE_V_BEGINACE]
2423 2857 5 DO
2424 2858 5 BEGIN
2425 2859 5 AED_L_FIRSTLINE = .AED_L_FIRSTLINE[LINE_L_BLINK];
```

ACT_UP - move up to previous line

1 16

15-Sep-1984 23:47:14

14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742

[ACLEDT.SRC]AEDMAIN.B32;1

Page 88

(19)

```
2426 2860 5      AED_W_TOTALLSIZE = .AED_W_TOTALLSIZE + .AED_L_FIRSTLINE[LINE_W_SIZE];
2427 2861 4      END;
2428 2862 4      AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
2429 2863 3      END;
2430 2864 3      BUFFER_INDEX = MIN (.SEGMENT_SIZE, .AED_B_COLUMN - 1);
2431 2865 2      END
2432 2866 2      ELSE AED_B_LINE = 1;
2433 2867 2      AED_SET_CURSOR (.AED_B_LINE, .BUFFER_INDEX + 1);
2434 2868 2      AED_L_FLAGS[AED_V_ENDACL] = 0;
2435 2869 2      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2436 2870 2      AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2437 2871 2      TERM_CHAR = 0;
2438 2872 2      RETURN 1;
2439 2873 2
2440 2874 1      END;
```

! End of routine ACT_UP

			001C 00000	ACT_UP:	.WORD	Save R2,R3,R4	2747
	54	0000'	CF 9E 00002		MOVAB	NEW_TEXT_LINE, R4	
	53	0000'	CF 9E 00007		MOVAB	AED_L_FIRSTLINE, R3	
	C1 A3	40 8F 8A 0000C			BICB2	#64, AED_L_FLAGS+1	2789
	50	F0 A3 9E 00011			MOVAB	AED_Q_LINETABLE, R0	2791
	50	74 A3 D1 00015			CMPL	AED_T_CURLINE+4, R0	
		03 12 00019			BNEQ	1\$	
		013C 31 0001B			BRW	18\$	
	0000G CF	00 FB 0001E	1\$:		CALLS	#0, AED_REPSEGMENT	2794
	64	50 D0 00023			MOVL	R0, NEW_TEXT_LINE	
	03	0A A0 E8 00026			BLBS	10(R0), 3\$	2795
		00AF 31 0002A	2\$:		BRW	11\$	
		C0 A3 95 0002D	3\$:		TSTB	AED_L_FLAGS	2798
		0A 19 00030			BLSS	4\$	
05	C1 A3	05 E0 00032			BBS	#5, AED_L_FLAGS+1, 4\$	2799
EE	C1 A3	06 E1 00037			BBC	#6, AED_L_FLAGS+1, 2\$	2800
	0000V CF	00 FB 0003C	4\$:		CALLS	#0, FINISH_ACE	2803
		C1 A3 95 00041			TSTB	AED_L_FLAGS+1	2804
		10 18 00044			BGEQ	5\$	
0B	C1 A3	04 E1 00046			BBC	#4, AED_L_FLAGS+1, 5\$	2805
	50	64 D0 0004B			MOVL	NEW_TEXT_LINE, R0	2808
	0A A0	04 88 0004E			BISB2	#4, 10(R0)	
		0284 C3 B4 00052			CLRW	AED_W_TOTALLSIZE	2809
	C1 A3	40 8F 8A 00056	5\$:		BICB2	#64, AED_L_FLAGS+1	2811
		0284 C3 B5 0005B			TSTW	AED_W_TOTALLSIZE	2812
		03 12 0005F			BNEQ	6\$	
	74	94 D0 00061			MOVL	@NEW_TEXT_LINE, NEW_TEXT_LINE	2813
	0000G CF	00 FB 00064	6\$:		CALLS	#0, AED_COMPRESS	2814
	7E	0284 C3 3C 00069			MOVZWL	AED_W_TOTALLSIZE, -(SP)	2815
	0000G CF	01 FB 0006E			CALLS	#1, AED_UPDATEACL	
	4C A3	50 D0 00073			MOVL	R0, AED_L_STATUS	
	5B	4C A3 E8 00077			BLBS	AED_L_STATUS, 10\$	2816
	C0 A3	40 8F 88 0007B			BISB2	#64, AED_L_FLAGS	2819
		63 DD 00080			PUSHL	AED_L_FIRSTLINE	2820
	0000G CF	01 FB 00082			CALLS	#1, AED_POSITION	
		63 DD 00087			PUSHL	AED_L_FIRSTLINE	2821
	0000G CF	01 FB 00089			CALLS	#1, AED_COPSEGMENT	

ACT_UP - move up to previous line

J 16
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32:1Page 89
(19)

	50		63	D0	0008E	MOVL	AED_L_FIRSTLINE, R0	2823
04	B0	70	A3	0E	00091	INSQUE	AED_T_CURLINE, 24(R0)	
	63	04	A3	D1	00096	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	2824
			05	12	0009A	BNEQ	7\$	
04	A3	70	A3	9E	0009C	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	2825
	63	08	A3	D1	000A1	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	2826
			05	12	000A5	BNEQ	8\$	
08	A3	70	A3	9E	000A7	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	2827
	63	70	A3	9E	000AC	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2828
04	A3		63	D1	000B0	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	2829
			09	13	000B4	BEQL	9\$	
04	C0		05	E1	000B6	BBC	#5, AED_L_FLAGS, 9\$	2830
	C0		20	8A	000BB	BICB2	#32, AED_C_FLAGS	2831
		E8	A4	D4	000BF	CLRL	BUFFER_INDEX	2832
E0	A3		01	90	000C2	MOVB	#1, AED_B_COLUMN	2833
	7E	E0	A3	9A	000C6	MOVZBL	AED_B_COLUMN, -(SP)	2834
	7E	E4	A3	9A	000CA	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	000CE	CALLS	#2, AED_SET_CURSOR	
			009A	31	000D3	BRW	20\$	2835
C0	A3	2080	BF	AA	000D6	BICW2	#8320, AED_L_FLAGS	2840
	50		64	D0	000DC	MOVL	NEW TEXT LINE, R0	2843
	52	04	A0	D0	000DF	MOVL	4(R0), UP LINE SEGMENT	
			52	DD	000E3	PUSHL	UP LINE SEGMENT	2844
0000G	CF		01	FB	000E5	CALLS	#1, AED_POSITION	
			52	DD	000EA	PUSHL	UP LINE SEGMENT	2845
0000G	CF		01	FB	000EC	CALLS	#1, AED_COPSEGMENT	
04	B2	70	A3	0E	000F1	INSQUE	AED_T_CURLINE, 24(UP LINE SEGMENT)	2846
	52	08	A3	D1	000F6	CMPL	AED_L_BEGINLINE, UP LINE SEGMENT	2847
			05	12	000FA	BNEQ	12\$	
08	A3	70	A3	9E	000FC	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	2848
	52		63	D1	00101	CMPL	AED_L_FIRSTLINE, UP LINE SEGMENT	2849
			04	12	00104	BNEQ	13\$	
33	63	70	A3	9E	00106	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2850
	A3		01	E1	0010A	BBC	#1, AED_T_CURLINE+TO, 16\$	2851
	50	70	A3	9E	0010F	MOVAB	AED_T_CURLINE, R0	2854
04	A3		50	D0	00113	MOVL	R0, AED_L_LASTLINE	
	63		50	D0	00117	MOVL	R0, AED_L_FIRSTLINE	
	50	04	A3	D0	0011A	MOVL	AED_L_LASTLINE, R0	2855
0284	C3	08	A0	B0	0011E	MOVW	8(R0), AED_W_TOTALSIZE	
	50		63	D0	00124	MOVL	AED_L_FIRSTLINE, R0	2856
	0F	0A	A0	E8	00127	BLBS	10(R0), 15\$	
	63	04	A0	D0	0012B	MOVL	4(R0), AED_L_FIRSTLINE	2859
	50		63	D0	0012F	MOVL	AED_L_FIRSTLINE, R0	2860
0284	C3	08	A0	A0	00132	ADDW2	8(R0), AED_W_TOTALSIZE	
			ED	11	00138	BRB	14\$	2856
	50		63	D0	0013A	MOVL	AED_L_FIRSTLINE, R0	2862
FC	A3	0C	A0	D0	0013D	MOVL	12(R0), AED_L_CURACE	
	51	E0	A3	9A	00142	MOVZBL	AED_B_COLUMN, R1	2864
			51	D7	00146	DECL	R1	
	50	78	A3	3C	00148	MOVZWL	SEGMENT_SIZE, R0	
	51		50	D1	0014C	CMPL	R0, R1	
			03	15	0014F	BLEQ	17\$	
	50		51	D0	00151	MOVL	R1, R0	
E8	A4		50	D0	00154	MOVL	R0, BUFFER_INDEX	
			04	11	00158	BRB	19\$	2791
	E4		01	90	0015A	MOVB	#1, AED_B_LINE	2866
7E	E8		01	C1	0015E	ADDL3	#1, BUFFER_INDEX, -(SP)	2867

AED\$MAIN
V04-000

ACT_UP - move up to previous line

K 16
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 90
(19)

0000G	7E	E4	A3	9A	00163	MOVZBL	AED_B LINE, -(SP)	:
C0	CF		D2	FB	00167	CALLS	#2, AED SET CURSOR	:
C1	A3		20	8A	0016C	BICB2	#32, AED_L_FLAGS	: 2868
	A3	2008	BF	AA	00170	BICW2	#8200, AED_L_FLAGS+1	: 2870
		10	A4	94	00176	CLRB	TERM CHAR	: 2871
	50		01	D0	00179	MOVL	#1, R0	: 2872
			04	0017C	RET			: 2874

; Routine Size: 381 bytes, Routine Base: \$CODE\$ + 1994

ACT_DOWN - move down to next line

```
2442 2875 1 $SBTTL 'ACT_DOWN - move down to next line'
2443 2876 1 ROUTINE ACT_DOWN =
2444 2877 1
2445 2878 1 ++
2446 2879 1
2447 2880 1 FUNCTIONAL DESCRIPTION:
2448 2881 1
2449 2882 1 This routine moves the cursor down to the next line segment. The
2450 2883 1 cursor position in the new line segment is minimized with the
2451 2884 1 current cursor position and the end of the line segment.
2452 2885 1
2453 2886 1 CALLING SEQUENCE:
2454 2887 1 ACT_DOWN ()
2455 2888 1
2456 2889 1 INPUT PARAMETERS:
2457 2890 1 none
2458 2891 1
2459 2892 1 IMPLICIT INPUTS:
2460 2893 1 OWN storage
2461 2894 1
2462 2895 1 OUTPUT PARAMETERS:
2463 2896 1 none
2464 2897 1
2465 2898 1 IMPLICIT OUTPUTS:
2466 2899 1 none
2467 2900 1
2468 2901 1 ROUTINE VALUE:
2469 2902 1 1 if successful
2470 2903 1 error status otherwise
2471 2904 1
2472 2905 1 SIDE EFFECTS:
2473 2906 1 The line segment table is updated as necessary, ACE line pointers
2474 2907 1 are updated, and the object's ACL is updated as necessary.
2475 2908 1
2476 2909 1 --
2477 2910 1
2478 2911 2 BEGIN
2479 2912 2
2480 2913 2 LOCAL
2481 2914 2 DOWN_LINE_SEGMENT : REF $BBLOCK; ! Address of next line segment
2482 2915 2
2483 2916 2 IF NOT .AED_L_FLAGS[AED_V_ENDACL]
2484 2917 2 OR .AED_W_TOTALSIZE GTR 0 OR .SEGMENT_SIZE GTR 0
2485 2918 2 THEN
2486 2919 2 BEGIN
2487 2920 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
2488 2921 2 IF .AED_C_LASTLINE EQLA NEW_TEXT_LINE[LINE_L_FLINK]
2489 2922 2 THEN
2490 2923 2 BEGIN
2491 2924 2 IF .AED_L_FLAGS[AED_V_MODIFIED]
2492 2925 2 OR .AED_L_FLAGS[AED_V_INSERT]
2493 2926 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
2494 2927 2 THEN
2495 2928 2 BEGIN
2496 2929 2 FINISH ACE ();
2497 2930 2 IF .AED_L_FLAGS[AED_V_PROMPT]
2498 2931 2 AND .AED_C_FLAGS[AED_V_FIRSTCHAR]
```

```
2499 2932 5 THEN
2500 2933 6 BEGIN
2501 2934 6 NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
2502 2935 6 AED_W_TOTALSIZE = 0;
2503 2936 6 END;
2504 2937 5 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2505 2938 5 IF .AED_W_TOTALSIZE EQL 0
2506 2939 5 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_BLINK];
2507 2940 5 AED_COMPRESS ();
2508 2941 5 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
2509 2942 5 IF NOT .AED_L_STATUS
2510 2943 5 THEN
2511 2944 6 BEGIN
2512 2945 6 AED_L_FLAGS[AED_V_ACERROR] = 1;
2513 2946 6 AED_POSITION (.AED_L_FIRSTLINE);
2514 2947 6 AED_COPSEGMENT (.AED_L_FIRSTLINE);
2515 2948 6 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2516 2949 6 .AED_C_FIRSTLINE[LINE_L_BLINK]);
2517 2950 6 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
2518 2951 6 THEN AED_C_LASTLINE = AED_T_CURLINE;
2519 2952 6 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
2520 2953 6 THEN AED_C_BEGINLINE = AED_T_CURLINE;
2521 2954 6 AED_L_FIRSTLINE = AED_T_CURLINE;
2522 2955 6 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
2523 2956 6 AND .AED_C_FLAGS[AED_V_ENDACE]
2524 2957 6 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
2525 2958 6 BUFFER_INDEX = 0;
2526 2959 6 AED_B_COLUMN = 1;
2527 2960 6 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2528 2961 6 AED_L_FLAGS[AED_V_GOLDREY] = 0;
2529 2962 6 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2530 2963 6 TERM_CHAR = 0;
2531 2964 6 RETURN 1;
2532 2965 6 END;
2533 2966 5 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
2534 2967 5 END;
2535 2968 5 END;
2536 2969 5 DOWN_LINE_SEGMENT = .NEW_TEXT_LINE[LINE_L_FLINK];
2537 2970 5 AED_POSITION (.DOWN_LINE_SEGMENT);
2538 2971 5 AED_COPSEGMENT (.DOWN_LINE_SEGMENT);
2539 2972 5 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .DOWN_LINE_SEGMENT[LINE_L_BLINK]);
2540 2973 5 IF .AED_L_LASTLINE EQL .DOWN_LINE_SEGMENT
2541 2974 5 THEN AED_C_LASTLINE = AED_T_CURLINE[LINE_L_FLINK];
2542 2975 5 IF .AED_C_BEGINLINE EQL .DOWN_LINE_SEGMENT
2543 2976 5 THEN AED_C_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
2544 2977 5 IF .DOWN_LINE_SEGMENT NEQ .AED_Q_LINETABLE[LINE_L_FLINK]
2545 2978 5 THEN
2546 2979 6 BEGIN
2547 2980 6 IF .AED_T_CURLINE[LINE_V_BEGINACE]
2548 2981 6 THEN
2549 2982 7 BEGIN
2550 2983 7 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
2551 2984 7 AED_W_TOTALSIZE = .AED_C_FIRSTLINE[LINE_Q_SIZE];
2552 2985 7 UNTIL .AED_L_LASTLINE[LINE_V_ENDACE]
2553 2986 7 DO
2554 2987 8 BEGIN
2555 2988 8 IF .AED_L_LASTLINE EQLA AED_T_CURLINE
```

ACT_DOWN - move down to next line

```
2556 2989 6 THEN AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
2557 2990 6 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
2558 2991 6 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
2559 2992 6 END;
2560 2993 5 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
2561 2994 4 END;
2562 2995 4 BUFFER_INDEX = MIN (.SEGMENT_SIZE, .AED_B_COLUMN - 1);
2563 2996 4 AED_SET_CURSOR (.AED_B_LINE, .BUFFER_INDEX + 1);
2564 2997 4 END;
2565 2998 3 ELSE
2566 2999 4 BEGIN
2567 3000 4 AED_L_FLAGS[AED_V_ENDACL] = 1;
2568 3001 4 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
2569 3002 4 AED_W_TOTALSIZE = .SEGMENT_SIZE = 0;
2570 3003 4 BUFFER_INDEX = 0;
2571 3004 4 AED_B_COLUMN = 1;
2572 3005 4 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
2573 3006 4 AED_L_FIRSTLINE[LINE_W_FLAGS] = LINE_M_BEGINACE;
2574 3007 4 AED_L_CURACE = 0;
2575 3008 4 IF .AED_L_FLAGS[AED_V_PROMPT]
2576 3009 4 THEN
2577 3010 3 BEGIN
2578 3011 3 AED_B_ACETYPE = 0;
2579 3012 3 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
2580 3013 3 AED_SELECTFIELD(.BUFFER_INDEX);
2581 3014 3 ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
2582 3015 3 ECHO_DESC[DSC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
2583 3016 3 SCR$SET_CURSOR (.AED_B_LINE, 1);
2584 3017 3 AED_PUTOUTPUT (ECHO_DESC);
2585 3018 3 SCR$ERASE_LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
2586 3019 3 AED_B_COLUMN = .BUFFER_INDEX + 1;
2587 3020 3 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2588 3021 3 AED_L_FLAGS[AED_V_FIRSTCHAR] = 1;
2589 3022 3 END;
2590 3023 3 END;
2591 3024 2 END;
2592 3025 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2593 3026 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2594 3027 2 TERM_CHAR = 0;
2595 3028 2 RETURN 1;
2596 3029 2
2597 3030 1 END;
```

! End of routine ACT_DOWN

001C 0000 ACT_DOWN:

54	0000'	CF	9E	00002	.WORD	Save R2,R3,R4	2876
53	0000'	CF	9E	00007	MOVAB	NEW_TEXT_LINE, R4	
63		05	E1	0000C	MOVAB	AED_L_FLAGS, R3	
	02C4	C3	B5	00010	BBC	#5, AED_L_FLAGS, 1\$	2916
		09	12	00014	TSTW	AED_W_TOTALSIZE	2917
	00B8	C3	B5	00016	BNEQ	1\$	
		03	12	0001A	TSTW	SEGMENT_SIZE	
		01F4	31	0001C	BNEQ	1\$	
					BRW	21\$	

ACT_DOWN - move down to next line

15-Sep-1984 23:47:14
14-Sep-1984 11:52:29X-11 Bliss-32 V4.0-742
LACLEDT.SRCJAEDMAIN.B32;1Page 94
(20)

	0000G	CF		00	FB	0001F	18:	CALLS	#0, AED_REPSEGMENT	2920
		64		50	DO	00024		MOVL	RO, NEW_TEXT_LINE	
		64	44	A3	D1	00027		CMPL	AED_L_LASTLINE, NEW_TEXT_LINE	2921
				03	13	0002B		BEQL	38	
				00B2	31	0002D	28:	BRW	118	
				63	95	00030	38:	TSTB	AED_L_FLAGS	2924
				0A	19	00032		BLSS	48	
05	01	A3		05	E0	00034		BBS	#5, AED_L_FLAGS+1, 48	2925
EF	01	A3		06	E1	00039		BBC	#6, AED_L_FLAGS+1, 28	2926
	0000V	CF		00	FB	0003E	48:	CALLS	#0, FINISH_ACE	2929
			01	A3	95	00043		TSTB	AED_L_FLAGS+1	2930
				10	18	00046		BGEQ	58	
0B	01	A3		04	E1	00048		BBC	#4, AED_L_FLAGS+1, 58	2931
		50		64	DO	0004D		MOVL	NEW_TEXT_LINE, RO	2934
	0A	A0		04	88	00050		BISB2	#4, -10(RO)	
			02C4	C3	B4	00054		CLRW	AED_W_TOTALSIZE	2935
	01	A3	40	8F	8A	00058	58:	BICB2	#64, AED_L_FLAGS+1	2937
			02C4	C3	B5	0005D		TSTW	AED_W_TOTALSIZE	2938
				07	12	00061		BNEQ	68	
		50		64	DO	00063		MOVL	NEW_TEXT_LINE, RO	2939
		64	04	A0	DO	00066		MOVL	4(RO), NEW_TEXT_LINE	
	0000G	CF		00	FB	0006A	68:	CALLS	#0, AED_COMPRESS	2940
		7E	02C4	C3	3C	0006F		MOVZWL	AED_W_TOTALSIZE, -(SP)	2941
	0000G	CF		01	FB	00074		CALLS	#1, AED_UPDATEACL	
	008C	C3		50	DO	00079		MOVL	RO, AED_L_STATUS	
		5A	008C	C3	E8	0007E		BLBS	AED_L_STATUS, 108	2942
		63	40	8F	88	00083		BISB2	#64, AED_L_FLAGS	2945
			40	A3	DD	00087		PUSHL	AED_L_FIRSTLINE	2946
	0000G	CF		01	FB	0008A		CALLS	#1, AED_POSITION	
			40	A3	DD	0008F		PUSHL	AED_L_FIRSTLINE	2947
	0000G	CF		01	FB	00092		CALLS	#1, AED_COPSEGMENT	
		50	40	A3	DO	00097		MOVL	AED_L_FIRSTLINE, RO	2949
	04	B0	00B0	C3	0E	0009B		INSQUE	AED_T_CURLINE, 24(RO)	
	40	A3	44	A3	D1	000A1		CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	2950
				06	12	000A6		BNEQ	78	
	44	A3	00B0	C3	9E	000AB		MOVAB	AED_T_CURLINE, AED_L_LASTLINE	2951
	40	A3	48	A3	D1	000AE	78:	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	2952
				06	12	000B3		BNEQ	88	
	48	A3	00B0	C3	9E	000B5		MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	2953
	40	A3	00B0	C3	9E	000BB	88:	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	2954
	44	A3	40	A3	D1	000C1		CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	2955
				07	13	000C6		BEQL	98	
03		63		05	E1	000C8		BBC	#5, AED_L_FLAGS, 98	2956
		63		20	8A	000CC		BICB2	#32, AED_L_FLAGS	2957
			E8	A4	D4	000CF	98:	CLRL	BUFFER_INDEX	2958
	20	A3		01	90	000D2		MOVB	#1, AED_B_COLUMN	2959
		7E	20	A3	9A	000D6		MOVZBL	AED_B_COLUMN, -(SP)	2960
				00A3	31	000DA		BRW	198	
		63	20B0	8F	AA	000DD	108:	BICW2	#8320, AED_L_FLAGS	2966
		52	00	B4	DO	000E2	118:	MOVL	NEW_TEXT_LINE, DOWN_LINE_SEGMENT	2969
				52	DD	000E6		PUSHL	DOWN_LINE_SEGMENT	2970
	0000G	CF		01	FB	000E8		CALLS	#1, AED_POSITION	
				52	DD	000ED		PUSHL	DOWN_LINE_SEGMENT	2971
	0000G	CF		01	FB	000EF		CALLS	#1, AED_COPSEGMENT	
	04	B2	00B0	C3	0E	000F4		INSQUE	AED_T_CURLINE, 24(DOWN_LINE_SEGMENT)	2972
		52	44	A3	D1	000FA		CMPL	AED_L_LASTLINE, DOWN_LINE_SEGMENT	2973
				06	12	000FE		BNEQ	128	

ACT_DOWN - move down to next line

D 1
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1Page 95
(20)

44	A3	00B0	C3	9E	00100	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	2974
	52	48	A3	D1	00106	CMPL	AED_L_BEGINLINE, DOWN_LINE_SEGMENT	2975
			06	12	0010A	BNEQ	13\$	
48	A3	00B0	C3	9E	0010C	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	2976
	50	30	A3	9E	00112	MOVAB	AED_Q_LINETABLE, R0	2977
	50		52	D1	00116	CMPL	DOWN_CINE_SEGMENT, R0	
			71	13	00119	BEQL	20\$	
	44	00BA	C3	E9	0011B	BLBC	AED_T_CURLINE+10, 17\$	2980
	50	00B0	C3	9E	00120	MOVAB	AED_T_CURLINE, R0	2983
44	A3		50	D0	00125	MOVL	R0, AED_L_LASTLINE	
40	A3		50	D0	00129	MOVL	R0, AED_L_FIRSTLINE	
	52	40	A3	D0	0012D	MOVL	AED_L_FIRSTLINE, R2	2984
02C4	C3	08	A2	B0	00131	MOVW	8(R2), AED_W_TOTALSIZE	
	51	44	A3	D0	00137	MOVL	AED_L_LASTLINE, R1	2985
1F	0A		01	E0	0013B	BBS	#1, 10(R1), 16\$	
	50	00B0	C3	9E	00140	MOVAB	AED_T_CURLINE, R0	2988
	50		51	D1	00145	CMPL	R1, R0	
			04	12	00148	BNEQ	15\$	
44	A3		61	D0	0014A	MOVL	(R1), AED_L_LASTLINE	2989
44	A3	44	B3	D0	0014E	MOVL	AED_L_LASTLINE, AED_L_LASTLINE	2990
	51	44	A3	D0	00153	MOVL	AED_L_LASTLINE, R1	2991
02C4	C3	08	A1	A0	00157	ADDW2	8(R1), AED_W_TOTALSIZE	
			DC	11	0015D	BRB	14\$	2985
3C	A3	0C	A2	D0	0015F	MOVL	12(R2), AED_L_CURACE	2993
	51	20	A3	9A	00164	MOVZBL	AED_B_COLUMN, R1	2995
			51	D7	00168	DECL	R1	
	50	00B8	C3	3C	0016A	MOVZWL	SEGMENT_SIZE, R0	
	51		50	D1	0016F	CMPL	R0, R1	
			03	15	00172	BLEQ	18\$	
	50		51	D0	00174	MOVL	R1, R0	
7E	E8		50	D0	00177	MOVL	R0, BUFFER_INDEX	
	E8		01	C1	0017B	ADDL3	#1, BUFFER_INDEX, -(SP)	2996
	7E	24	A3	9A	00180	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	00184	CALLS	#2, AED_SET_CURSOR	
			0087	31	00189	BRW	21\$	2977
	63	4020	8F	A8	0018C	BISW2	#16416, AED_L_FLAGS	3001
		00B8	C3	B4	00191	CLRW	SEGMENT_SIZE	3002
		02C4	C3	B4	00195	CLRW	AED_W_TOTALSIZE	
		E8	A4	D4	00199	CLRL	BUFFER_INDEX	3003
20	A3		01	90	0019C	MOVB	#1, AED_B_COLUMN	3004
	50	00B0	C3	9E	001A0	MOVAB	AED_T_CURLINE, R0	3005
44	A3		50	D0	001A5	MOVL	R0, AED_L_LASTLINE	
40	A3		50	D0	001A9	MOVL	R0, AED_L_FIRSTLINE	
0A	A0		01	B0	001AD	MOVW	#1, 10(R0)	3006
		3C	A3	D4	001B1	CLRL	AED_L_CURACE	3007
		01	A3	95	001B4	TSTB	AED_L_FLAGS+1	3008
			5A	18	001B7	BGEQ	21\$	
		00A8	C3	9A	001B9	CLRB	AED_B_ACETYPE	3011
02	A3		08	8A	001BD	BICB2	#8, AED_L_FLAGS+2	3012
		E8	A4	9F	001C1	PUSHAB	BUFFER_INDEX	3013
0000G	CF		01	FB	001C4	CALLS	#1, AED_SELECTFIELD	
EC	A4	00B8	C3	B0	001C9	MOVW	AED_T_CURLINE+8, ECHO_DESC	3014
FO	A4	00C4	C3	9E	001CF	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	3015
			01	D0	001D5	PUSHL	#1	3016
	7E	24	A3	9A	001D7	MOVZBL	AED_B_LINE, -(SP)	
00000000G	00		02	FB	001DB	CALLS	#2, SCRSET_CURSOR	
		EC	A4	9F	001E2	PUSHAB	ECHO_DESC	3017

AEDSMAN
V04-000

ACT_DOWN - move down to next line

E 1
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDY.SRC]AEDMAIN.B32;1

Page 96
(20)

	0000G	CF		01	FB	001E5	CALLS	#1, AED_PUTOUTPUT	...	
		7E	00B8	C3	3C	001EA	MOVZWL	SEGMENT_SIZE, -(SP)	...	3018
				6E	D6	001EF	INCL	(SP)	...	
		7E	24	A3	9A	001F1	MOVZBL	AED_B_LINE, -(SP)	...	
20	A3	00000000G		02	FB	001F5	CALLS	#2, SCRSEASE LINE	...	3019
		E8		01	81	001FC	ADDB3	#1, BUFFER INDEX, AED_B_COLUMN	...	3020
				A3	9A	00202	MOVZBL	AED_B_COLUMN, -(SP)	...	
			20	A3	9A	00206	MOVZBL	AED_B_LINE, -(SP)	...	
			24	A3	9A	0020A	CALLS	#2, AED SET CURSOR	...	
	0000G	CF		02	FB	0020A	BISB2	#16, AED_L_FLAGS+1	...	3021
	01	A3		10	88	0020F	BICW2	#8200, AED_L_FLAGS+1	...	3026
	01	A3	2008	8F	AA	00213	CLRB	TERM CHAR	...	3027
			10	A4	94	00219	MOVL	#1, R0	...	3028
				01	D0	0021C	RET		...	3030
				04	00	0021F			...	

; Routine Size: 544 bytes, Routine Base: \$CODE\$ + 1B11

ACT_RIGHT - move right one character

```
2599 3031 1 XSBTTL 'ACT_RIGHT - move right one character'
2600 3032 1 ROUTINE ACT_RIGHT =
2601 3033 1
2602 3034 1 ++
2603 3035 1
2604 3036 1 FUNCTIONAL DESCRIPTION:
2605 3037 1
2606 3038 1 This routine advances the cursor one character to the right. If
2607 3039 1 the end of the line segment is reached, the cursor is set to the
2608 3040 1 first character of the next line.
2609 3041 1
2610 3042 1 CALLING SEQUENCE:
2611 3043 1 ACT_RIGHT ()
2612 3044 1
2613 3045 1 INPUT PARAMETERS:
2614 3046 1 none
2615 3047 1
2616 3048 1 IMPLICIT INPUTS:
2617 3049 1 OWN storage
2618 3050 1
2619 3051 1 OUTPUT PARAMETERS:
2620 3052 1 none
2621 3053 1
2622 3054 1 IMPLICIT OUTPUTS:
2623 3055 1 none
2624 3056 1
2625 3057 1 ROUTINE VALUE:
2626 3058 1 1 if successful
2627 3059 1 error status otherwise
2628 3060 1
2629 3061 1 SIDE EFFECTS:
2630 3062 1 The line segment table is updated as necessary, ACE line pointers
2631 3063 1 are updated, and the object's ACL is updated as necessary.
2632 3064 1
2633 3065 1 --
2634 3066 1
2635 3067 1 BEGIN
2636 3068 1
2637 3069 1 IF .BUFFER_INDEX LSS .SEGMENT_SIZE
2638 3070 1 THEN
2639 3071 1 BEGIN
2640 3072 1 BUFFER_INDEX = .BUFFER_INDEX + 1;
2641 3073 1 AED_B_COLUMN = .BUFFER_INDEX + 1;
2642 3074 1 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2643 3075 1 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2644 3076 1 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2645 3077 1 TERM_CHAR = 0;
2646 3078 1 RETURN 1;
2647 3079 1 END;
2648 3080 1
2649 3081 1 BUFFER_INDEX = 0;
2650 3082 1 AED_B_COLUMN = 1;
2651 3083 1 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2652 3084 1 TERM_CHAR = KEY_C_DOWN;
2653 3085 1 RETURN 1;
2654 3086 1
2655 3087 1 END;
```

! End of routine ACT_RIGHT

				000C 00000 ACT_RIGHT:				
			53	0000'	CF 9E 00002	WORD	Save R2, R3	3032
			52	0000'	CF 9E 00007	MOVAB	BUFFER_INDEX, R3	
63	0098	C2	10		00 ED 0000C	MOVAB	AED_B_COLUMN, R2	
					1D 15 00013	CMPZV	#0, #16, SEGMENT_SIZE, BUFFER_INDEX	3069
					63 D6 00015	BLEQ	18	
					01 81 00017	INCL	BUFFER_INDEX	3072
		62	63		62 9A 0001B	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	3073
			7E		A2 9A 0001E	MOVZBL	AED_B_COLUMN, -(SP)	3074
			7E	04	02 FB 00022	MOVZBL	AED_B_LINE, -(SP)	
	0000G		CF		8F AA 00027	CALLS	#2, AED SET CURSOR	
	E1		A2	2008	A3 94 0002D	BICW2	#8200, AED_L_FLAGS+1	3076
				28	0D 11 00030	CLRB	TERM_CHAR	3077
					63 D4 00032	BRB	28	3078
					01 90 00034	CLRL	BUFFER_INDEX	3081
			62		08 8A 00037	MOVB	#1, AED_B_COLUMN	3082
	E1		A2		1C 90 0003B	BICB2	#8, AED_L_FLAGS+1	3083
	28		A3		01 D0 0003F	MOVB	#28, TERM_CHAR	3084
			50		04 00042	MOVL	#1, R0	3085
						RET		3087

; Routine Size: 67 bytes, Routine Base: \$CODE\$ + 1031

ACT_LEFT - move left one character

```
2657 3088 1 XSBTTL 'ACT_LEFT - move left one character'
2658 3089 1 ROUTINE ACT_LEFT =
2659 3090 1
2660 3091 1 ++
2661 3092 1
2662 3093 1 FUNCTIONAL DESCRIPTION:
2663 3094 1
2664 3095 1 This routine advances the cursor one character to the left. If
2665 3096 1 the beginning of the line segment is reached, the cursor is set
2666 3097 1 to the last character of the previous line.
2667 3098 1
2668 3099 1 CALLING SEQUENCE:
2669 3100 1 ACT_LEFT ()
2670 3101 1
2671 3102 1 INPUT PARAMETERS:
2672 3103 1 none
2673 3104 1
2674 3105 1 IMPLICIT INPUTS:
2675 3106 1 OWN storage
2676 3107 1
2677 3108 1 OUTPUT PARAMETERS:
2678 3109 1 none
2679 3110 1
2680 3111 1 IMPLICIT OUTPUTS:
2681 3112 1 none
2682 3113 1
2683 3114 1 ROUTINE VALUE:
2684 3115 1 1 if successful
2685 3116 1 error status otherwise
2686 3117 1
2687 3118 1 SIDE EFFECTS:
2688 3119 1 The line segment table is updated as necessary, ACE line pointers
2689 3120 1 are updated, and the object's ACL is updated as necessary.
2690 3121 1
2691 3122 1 --
2692 3123 1
2693 3124 2 BEGIN
2694 3125 2
2695 3126 2 IF .BUFFER_INDEX GTR 0
2696 3127 2 THEN
2697 3128 2 BEGIN
2698 3129 2 BUFFER_INDEX = .BUFFER_INDEX - 1;
2699 3130 2 AED_B_COLUMN = .BUFFER_INDEX + 1;
2700 3131 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2701 3132 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2702 3133 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2703 3134 2 TERM_CHAR = 0;
2704 3135 2 RETURN 1;
2705 3136 2 END;
2706 3137 2
2707 3138 2 AED_B_COLUMN = .SBBLOCK [.AED_T_CURLINE[LINE_L_BLINK], LINE_W_SIZE] + 1;
2708 3139 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2709 3140 2 TERM_CHAR = KEY_C_UP;
2710 3141 2 RETURN 1;
2711 3142 2
2712 3143 1 END;

! End of routine ACT_LEFT
```

				000C 00000 ACT_LEFT:					
		53	0000'	CF	9E	00002	.WORD	Save R2,R3	3089
		52	0000'	CF	9E	00007	MOVAB	BUFFER_INDEX, R3	
				63	D5	0000C	MOVAB	AED_B_COLUMN, R2	
				1D	15	0000E	TSTL	BUFFER_INDEX	3126
				63	D7	00010	BLEQ	1\$	
62		63		01	81	00012	DECL	BUFFER_INDEX	3129
		7E		62	9A	00016	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	3130
		7E	04	A2	9A	00019	MOVZBL	AED_B_COLUMN, -(SP)	3131
	0000G	CF		02	FB	0001D	MOVZBL	AED_B_LINE, -(SP)	
	E1	A2	2008	0F	AA	00022	CALLS	#2, AED_SET_CURSOR	
			28	A3	94	00028	BICW2	#8200, AED_L_FLAGS+1	3133
				12	11	0002B	CLRB	TERM_CHAR	3134
		50	0094	C2	D0	0002D	BRB	2\$	3135
62	0B	A0		01	81	00032	MOVL	AED_T_CURLINE+4, R0	3138
	E1	A2		08	8A	00037	ADDB3	#1, -8TRO), AED_B_COLUMN	
	28	A3		1B	90	0003B	BICB2	#8, AED_L_FLAGS+T	3139
		50		01	D0	0003F	MOVB	#27, TERM_CHAR	3140
				04	00	00042	MOVL	#1, R0	3141
							RET		3143

; Routine Size: 67 bytes, Routine Base: \$CODE\$ + 1D74

ACT_TOP - move to beginning of ACL

```
2714 3144 ZSBTTL 'ACT_TOP - move to beginning of ACL'
2715 3145 ROUTINE ACT_TOP =
2716 3146
2717 3147 ++
2718 3148
2719 3149 FUNCTIONAL DESCRIPTION:
2720 3150
2721 3151     This routine moves the cursor to the first character in the first
2722 3152     line segment of the first ACE in the ACL. The display is scrolled
2723 3153     as necessary.
2724 3154
2725 3155 CALLING SEQUENCE:
2726 3156     ACT_TOP ()
2727 3157
2728 3158 INPUT PARAMETERS:
2729 3159     none
2730 3160
2731 3161 IMPLICIT INPUTS:
2732 3162     OWN storage
2733 3163
2734 3164 OUTPUT PARAMETERS:
2735 3165     none
2736 3166
2737 3167 IMPLICIT OUTPUTS:
2738 3168     none
2739 3169
2740 3170 ROUTINE VALUE:
2741 3171     1 if successful
2742 3172     error status otherwise
2743 3173
2744 3174 SIDE EFFECTS:
2745 3175     The line segment table is updated as necessary, ACE line pointers
2746 3176     are updated, and the object's ACL is updated as necessary.
2747 3177
2748 3178 --
2749 3179
2750 3180 BEGIN
2751 3181
2752 3182 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;                                ! No item selection
2753 3183
2754 3184 NEW_TEXT_LINE = AED_REPSEGMENT ();
2755 3185 IF .AED_C_FLAGS[AED_V_MODIFIED]
2756 3186 OR .AED_L_FLAGS[AED_V_INSERT]
2757 3187 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
2758 3188 THEN
2759 3189     BEGIN
2760 3190     FINISH ACE ();
2761 3191     IF .AED_L_FLAGS[AED_V_PROMPT]
2762 3192     AND .AED_C_FLAGS[AED_V_FIRSTCHAR]
2763 3193     THEN
2764 3194         BEGIN
2765 3195         NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
2766 3196         AED_W_TOTALSIZE = 0;
2767 3197         END;
2768 3198     AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2769 3199     IF .AED_W_TOTALSIZE EQL 0
2770 3200     THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
```

ACT_TOP - move to beginning of ACL

```
2771 3201 3 AED_COMPRESS ();
2772 3202 3 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
2773 3203 3 IF NOT .AED_L_STATUS
2774 3204 3 THEN
2775 3205 3 BEGIN
2776 3206 3 AED_L_FLAGS[AED_V_ACERROR] = 1;
2777 3207 3 AED_POSITION (.AED_L_FIRSTLINE);
2778 3208 3 AED_COPSEGMENT (.AED_L_FIRSTLINE);
2779 3209 3 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2780 3210 3 .AED_L_FIRSTLINE[LINE_L_BLINK]);
2781 3211 3 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
2782 3212 3 THEN AED_L_LASTLINE = AED_T_CURLINE;
2783 3213 3 IF .AED_L_BEGINLINE EQL .AED_L_FIRSTLINE
2784 3214 3 THEN AED_L_BEGINLINE = AED_T_CURLINE;
2785 3215 3 AED_L_FIRSTLINE = AED_T_CURLINE;
2786 3216 3 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
2787 3217 3 AND .AED_L_FLAGS[AED_V_ENDACL]
2788 3218 3 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
2789 3219 3 BUFFER_INDEX = 0;
2790 3220 3 AED_B_COLUMN = 1;
2791 3221 3 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2792 3222 3 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2793 3223 3 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2794 3224 3 TERM_CHAR = 0;
2795 3225 3 RETURN 1;
2796 3226 3 END;
2797 3227 3 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
2798 3228 3 END;
2799 3229 2 AED_COMPRESS ();
2800 3230 2 AED_L_FIRSTLINE = .AED_Q_LINETABLE[LINE_L_FLINK];
2801 3231 2 AED_POSITION (.AED_L_FIRSTLINE);
2802 3232 2 AED_COPSEGMENT (.AED_L_FIRSTLINE);
2803 3233 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_L_FIRSTLINE[LINE_L_BLINK]);
2804 3234 2 AED_L_BEGINLINE = AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE[LINE_L_FLINK];
2805 3235 2 AED_W_TOTALSIZE = .AED_L_LASTLINE[LINE_W_SIZE];
2806 3236 2 UNTIL .AED_L_LASTLINE[LINE_V_ENDACE]
2807 3237 2 DO
2808 3238 2 BEGIN
2809 3239 2 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
2810 3240 2 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
2811 3241 2 END;
2812 3242 2 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
2813 3243 2 BUFFER_INDEX = 0;
2814 3244 2 AED_B_LINE = AED_B_COLUMN = 1;
2815 3245 2 AED_SET_CURSOR (T, 1);
2816 3246 2 AED_L_FLAGS[AED_V_ENDACL] = 0;
2817 3247 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2818 3248 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2819 3249 2 TERM_CHAR = 0;
2820 3250 2 RETURN 1;
2821 3251 2
2822 3252 1 END;
```

! End of routine ACT_TOP

			0000	CF	9E	00002	ACT_TOP: .WORD	Save R2,R3	3145
			0000	CF	9E	00007	MOVAB	NEW_TEXT_LINE, R3	
	01	A2	40	8F	8A	0000C	MOVAB	AED_L_FLAGS, R2	
	0000G	CF		00	FB	00011	BICB2	#64, AED_L_FLAGS+1	3182
		63		50	D0	00016	CALLS	#0, AED_REPSEGMENT	3184
				62	95	00019	MOVL	R0, NEW_TEXT_LINE	
				0D	19	0001B	TSTB	AED_L_FLAGS	3185
08	01	A2		05	E0	0001D	BLSS	1\$	
03	01	A2		06	E0	00022	BBS	#5, AED_L_FLAGS+1, 1\$	3186
				00A8	31	00027	BBS	#6, AED_L_FLAGS+1, 1\$	3187
	0000V	CF		00	FB	0002A	BRW	8\$	
			01	A2	95	0002F	CALLS	#0, FINISH ACE	3190
				10	1B	00032	TSTB	AED_L_FLAGS+1	3191
08	01	A2		04	E1	00034	BGEQ	2\$	
	50			63	D0	00039	BBC	#4, AED_L_FLAGS+1, 2\$	3192
	0A	A0		04	88	0003C	MOVL	NEW_TEXT_LINE, R0	3195
			02C4	C2	B4	00040	BISB2	#4, -10(R0)	
	01	A2	40	BF	8A	00044	CLRW	AED_W_TOTALSIZE	3196
			02C4	C2	B5	00049	BICB2	#64, AED_L_FLAGS+1	3198
				03	12	0004D	TSTW	AED_W_TOTALSIZE	3199
				93	D0	0004F	BNEQ	3\$	
	0000G	CF		00	FB	00052	MOVL	NEW_TEXT_LINE, NEW_TEXT_LINE	3200
		7E	02C4	C2	3C	00057	CALLS	#0, AED_COMPRESS	3201
	0000G	CF		01	FB	0005C	MOVZWL	AED_W_TOTALSIZE, -(SP)	3202
	008C	C2		50	D0	00061	CALLS	#1, AED_UPDATEACL	
		62	008C	C2	E8	00066	MOVL	R0, AED_L_STATUS	
		62	40	8F	88	0006B	BLBS	AED_L_STATUS, 7\$	3203
			40	A2	DD	0006F	BISB2	#64, AED_L_FLAGS	3206
	0000G	CF		01	FB	00072	PUSHL	AED_L_FIRSTLINE	3207
			40	A2	DD	00077	CALLS	#1, AED_POSITION	
	0000G	CF		01	FB	0007A	PUSHL	AED_L_FIRSTLINE	3208
		50	40	A2	D0	0007F	CALLS	#1, AED_COPSEGMENT	
	04	B0	00B0	C2	0E	00083	MOVL	AED_L_FIRSTLINE, R0	3210
	40	A2	44	A2	D1	00089	INSQUE	AED_T_CURLINE, 24(R0)	
				06	12	0008E	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	3211
	44	A2	00B0	C2	9E	00090	BNEQ	4\$	
	40	A2	48	A2	D1	00096	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	3212
				06	12	0009B	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	3213
	48	A2	00B0	C2	9E	0009D	BNEQ	5\$	
	40	A2	00B0	C2	9E	000A3	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	3214
	44	A2	40	A2	D1	000A9	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	3215
				07	13	000AE	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	3216
03		62		05	E1	000B0	BEQL	6\$	
		62		20	8A	000B4	BBC	#5, AED_L_FLAGS, 6\$	3217
			E8	A3	D4	000B7	BICB2	#32, AED_L_FLAGS	3218
	20	A2		01	90	000BA	CLRL	BUFFER_INDEX	3219
		7E	20	A2	9A	000BE	MOVB	#1, AED_B_COLUMN	3220
		7E	24	A2	9A	000C2	MOVZBL	AED_B_COLUMN, -(SP)	3221
	0000G	CF		02	FB	000C6	MOVZBL	AED_B_LINE, -(SP)	
				7D	11	000CB	CALLS	#2, AED_SET_CURSOR	
		62	2080	BF	AA	000CD	BRB	11\$	3222
	0000G	CF		00	FB	000D2	BICW2	#8320, AED_L_FLAGS	3227
	40	A2	30	A2	D0	000D7	CALLS	#0, AED_COMPRESS	3229
			40	A2	DD	000DC	MOVL	AED_Q_LINETABLE, AED_L_FIRSTLINE	3230
	0000G	CF		01	FB	000DF	PUSHL	AED_L_FIRSTLINE	3231
			40	A2	DD	000E4	CALLS	#1, AED_POSITION	
							PUSHL	AED_L_FIRSTLINE	3232

0000G	CF		01	FB	000E7	CALLS	#1, AED COPSEGMENT	...	
	50	40	A2	D0	000EC	MOVL	AED_L_FIRSTLINE, R0	...	3233
04	B0	00B0	C2	0E	000F0	INSQUE	AED_T_CURLINE, 84(R0)	...	
	50	00B0	C2	9E	000F6	MOVAB	AED_T_CURLINE, R0	...	3234
44	A2		50	D0	000FB	MOVL	R0, AED_L_LASTLINE	...	
40	A2		50	D0	000FF	MOVL	R0, AED_L_FIRSTLINE	...	
48	A2		50	D0	00103	MOVL	R0, AED_L_BEGINLINE	...	
	50	44	A2	D0	00107	MOVL	AED_L_LASTLINE, R0	...	3235
02C4	C2	08	A0	B0	0010B	MOVW	8(R0), AED_W_TOTALSIZE	...	
	50	44	A2	D0	00111	MOVL	AED_L_LASTLINE, R0	...	3236
10	0A		01	E0	00115	BBS	#1, 10(R0), 10\$...	
	44		60	D0	0011A	MOVL	(R0), AED_L_LASTLINE	...	3239
	50	44	A2	D0	0011E	MOVL	AED_L_LASTLINE, R0	...	3240
02C4	C2	08	A0	30	00122	ADDW2	8(R0), AED_W_TOTALSIZE	...	
	50	40	EB	11	00128	BRB	9\$...	3236
	3C		A2	D0	0012A	MOVL	AED_L_FIRSTLINE, R0	...	3242
	A2	0C	A0	D0	0012E	MOVL	12(R0), AED_L_CURACE	...	
		EB	A3	D4	00133	CLRL	BUFFER INDEX	...	3243
20	A2		01	90	00136	MOVB	#1, AED_B_COLUMN	...	3244
24	A2		01	90	0013A	MOVB	#1, AED_B_LINE	...	
			01	DD	0013E	PUSHL	#1	...	3245
			01	DD	00140	PUSHL	#1	...	
0000G	CF		02	FB	00142	CALLS	#2, AED SET CURSOR	...	
	62		20	8A	00147	BICB2	#32, AED_L_FLAGS	...	3246
01	A2	2008	8F	AA	0014A	BICW2	#8200, AED_L_FLAGS+1	...	3248
		10	A3	94	00150	CLRB	TERM CHAR	...	3249
	50		01	D0	00153	MOVL	#1, R0	...	3250
			04	00	00156	RET		...	3252

; Routine Size: 343 bytes, Routine Base: \$CODE\$ + 10B7

ACT_BOTTOM - move to end of ACL

```
2824 3253 1 XSBTTL 'ACT_BOTTOM - move to end of ACL'
2825 3254 1 ROUTINE ACT_BOTTOM =
2826 3255 1
2827 3256 1 ++
2828 3257 1
2829 3258 1 FUNCTIONAL DESCRIPTION:
2830 3259 1
2831 3260 1     This routine positions the cursor to the first character position
2832 3261 1     in a new ACE at the end of the ACL. The screen is scrolled as
2833 3262 1     necessary.
2834 3263 1
2835 3264 1 CALLING SEQUENCE:
2836 3265 1     ACT_BOTTOM ()
2837 3266 1
2838 3267 1 INPUT PARAMETERS:
2839 3268 1     none
2840 3269 1
2841 3270 1 IMPLICIT INPUTS:
2842 3271 1     OWN storage
2843 3272 1
2844 3273 1 OUTPUT PARAMETERS:
2845 3274 1     none
2846 3275 1
2847 3276 1 IMPLICIT OUTPUTS:
2848 3277 1     none
2849 3278 1
2850 3279 1 ROUTINE VALUE:
2851 3280 1     1 if successful
2852 3281 1     error status otherwise
2853 3282 1
2854 3283 1 SIDE EFFECTS:
2855 3284 1     The line segment table is updated as necessary, ACE line pointers
2856 3285 1     are updated, and the object's ACL is updated as necessary.
2857 3286 1
2858 3287 1 --
2859 3288 1
2860 3289 2 BEGIN
2861 3290 2
2862 3291 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
2863 3292 2 IF .AED_L_FLAGS[AED_V_MODIFIED]
2864 3293 2 OR .AED_L_FLAGS[AED_V_INSERT]
2865 3294 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
2866 3295 2 THEN
2867 3296 2 BEGIN
2868 3297 2     FINISH ACE ();
2869 3298 2     IF .AED_L_FLAGS[AED_V_PROMPT]
2870 3299 2     AND .AED_L_FLAGS[AED_V_FIRSTCHAR]
2871 3300 2     THEN
2872 3301 4 BEGIN
2873 3302 4     NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
2874 3303 4     AED_W_TOTALSIZE = 0;
2875 3304 4     END;
2876 3305 4 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
2877 3306 4 IF .AED_W_TOTALSIZE EQL 0
2878 3307 4 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_BLINK];
2879 3308 4 AED_COMPRESS T);
2880 3309 4 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
```

```
2881 3310 3 IF NOT .AED_L_STATUS
2882 3311 3 THEN
2883 3312 4 BEGIN
2884 3313 4 AED_L_FLAGS[AED_V_ACERROR] = 1;
2885 3314 4 AED_POSITION (.AED_L_FIRSTLINE);
2886 3315 4 AED_COSEMENT (.AED_L_FIRSTLINE);
2887 3316 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
2888 3317 4 .AED_C_FIRSTLINE[LINE_L_BLINK]);
2889 3318 4 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
2890 3319 4 THEN AED_C_LASTLINE = AED_T_CURLINE;
2891 3320 4 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
2892 3321 4 THEN AED_C_BEGINLINE = AED_T_CURLINE;
2893 3322 4 AED_L_FIRSTLINE = AED_T_CURLINE;
2894 3323 4 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
2895 3324 4 AND .AED_C_FLAGS[AED_V_ENDACL]
2896 3325 4 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
2897 3326 4 BUFFER_INDEX = 0;
2898 3327 4 AED_B_COLUMN = 1;
2899 3328 4 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2900 3329 4 AED_L_FLAGS[AED_V_GODREY] = 0;
2901 3330 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2902 3331 4 TERM_CHAR = 0;
2903 3332 4 RETURN 1;
2904 3333 4 END;
2905 3334 3 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
2906 3335 3 END;
2907 3336 2 AED_COMPRESS ();
2908 3337 2 AED_POSITION (AED_Q_LINETABLE[LINE_L_FLINK]);
2909 3338 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_Q_LINETABLE[LINE_L_BLINK]);
2910 3339 2 AED_L_FLAGS[AED_V_ENDACL] = 1;
2911 3340 2 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
2912 3341 2 AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
2913 3342 2 BUFFER_INDEX = 0;
2914 3343 2 AED_B_COLUMN = 1;
2915 3344 2 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
2916 3345 2 AED_L_FIRSTLINE[LINE_Q_FLAGS] = LINE_M_BEGINACE;
2917 3346 2 AED_L_CURACE = 0;
2918 3347 2 IF .AED_L_FLAGS[AED_V_PROMPT]
2919 3348 2 THEN
2920 3349 3 BEGIN
2921 3350 3 AED_B_ACETYPE = 0;
2922 3351 3 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
2923 3352 3 AED_SELECTFIELD (BUFFER_INDEX);
2924 3353 3 ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
2925 3354 3 ECHO_DESC[DSC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
2926 3355 3 SCR$SET_CURSOR T.AED_B_LINE, 1;
2927 3356 3 AED_PUTOUTPUT (ECHO_DESC);
2928 3357 3 SCR$ERASE_LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
2929 3358 3 AED_B_COLUMN = .BUFFER_INDEX + 1;
2930 3359 3 AED_L_FLAGS[AED_V_FIRSTCHAR] = 1;
2931 3360 3 END;
2932 3361 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
2933 3362 2 AED_L_FLAGS[AED_V_GODREY] = 0;
2934 3363 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
2935 3364 2 TERM_CHAR = 0;
2936 3365 2 RETURN 1;
2937 3366 2
```


: 2938

3367 1 END;

! End of routine ACT_BOTTOM

			000C 00000 ACT_BOTTOM:			
		53	0000'	CF 9E 00002	.WORD	3254
		52	0000'	CF 9E 00007	MOVAB	
	0000G	CF		00 FB 0000C	MOVAB	
		63		50 D0 00011	CALLS	3291
				62 95 00014	MOVL	
				0D 19 00016	RO, NEW_TEXT_LINE	
				05 E0 00018	TSTB	3292
				06 E0 0001D	AED_L_FLAGS	
08	01	A2		00A0 31 00022	BLSS	
03	01	A2		00 00 FB 00025	1\$	
	0000V	CF		01 A2 95 0002A	BBS	3293
				10 18 0002D	BBS	3294
				04 E1 0002F	BRW	
				63 D0 00034	CALLS	3297
	0A	A0		04 88 00037	TSTB	3298
	01	A2	02C4	C2 B4 0003B	BGEQ	
			40	8F 8A 0003F	2\$	
			02C4	C2 B5 00044	BBC	3299
				07 12 00048	MOVAB	
		50		63 D0 0004A	MOVL	3302
		63	04	A0 D0 0004D	MOVL	
	0000G	CF		00 FB 00051	3\$	
		7E	02C4	C2 3C 00056	CALLS	3303
	0000G	CF		01 FB 0005B	MOVZWL	3305
	008C	C2		50 D0 00060	CALLS	3306
		56	008C	C2 E8 00065	MOVL	
		62	40	8F 88 0006A	BLBS	3310
			40	A2 DD 0006E	BISB2	3313
	0000G	CF		01 FB 00071	PUSHL	3314
			40	A2 DD 00076	CALLS	
	0000G	CF		01 FB 00079	CALLS	3315
		50		40 A2 D0 0007E	PUSHL	
	04	B0	00B0	C2 0E 00082	CALLS	3317
	40	A2	44	A2 D1 00088	MOVL	
				06 12 0008D	INSQUE	
	44	A2	00B0	C2 9E 0008F	CMPL	3318
	40	A2	48	A2 D1 00095	BNEQ	
				06 12 0009A	4\$	
	48	A2	00B0	C2 9E 0009C	MOVAB	3319
	40	A2	00B0	C2 9E 000A2	CMPL	3320
	44	A2	40	A2 D1 000A8	BNEQ	
				07 13 000AD	5\$	
		62		05 E1 000AF	MOVAB	3321
		62		20 8A 000B3	MOVAB	3322
			E8	A3 D4 000B6	CMPL	3323
	20	A2		01 90 000B9	BEQL	
			0092	31 000BD	6\$	
		62	2080	8F AA 000C0	7\$	
	0000G	CF		00 FB 000C5	8\$	
					CALLS	3324
						3325
						3326
						3327
						3328
						3334
						3336

		30	A2	9F	000CA	PUSHAB	AED_Q LINETABLE	...	3337
	0000G	CF	01	FB	000CD	CALLS	#1, AED_POSITION	...	
	34	B2	C2	0E	000D2	INSQUE	AED_T_CORLINE, @AED_Q_LINETABLE+4	...	3338
		62	8F	A8	000D8	BISW2	#16, AED_L_FLAGS	...	3340
		00B0	C2	B4	000DD	CLRW	SEGMENT_SIZE	...	3341
		4020	C2	B4	000E1	CLRW	AED_W_TOTALSIZE	...	
		00B8	C2	B4	000E1	CLRL	BUFFER_INDEX	...	3342
		02C4	A3	D4	000E5	MOVW	#1, AED_B_COLUMN	...	3343
		E8	01	90	000E8	MOVAB	AED_T_CORLINE, R0	...	3344
	20	A2	C2	9E	000EC	MOVL	R0, AED_L_LASTLINE	...	
		50	50	D0	000F1	MOVL	R0, AED_L_FIRSTLINE	...	
	44	A2	50	D0	000F5	MOVW	#1, 10(R0)	...	3345
	40	A2	01	B0	000F9	CLRL	AED_L_CURACE	...	3346
	0A	A0	A2	D4	000FD	TSTB	AED_L_FLAGS+1	...	3347
			01	A2	95	BGEQ	98	...	
		3C	C2	94	00105	CLRB	AED_B_ACETYPE	...	3350
		01	08	8A	00109	BICB2	#8, AED_L_FLAGS+2	...	3351
		00AB	A3	9F	0010D	PUSHAB	BUFFER_INDEX	...	3352
		E8	01	FB	00110	CALLS	#1, AED_SELECTFIELD	...	
	0000G	CF	C2	B0	00115	MOVW	AED_T_CORLINE+8, ECHO_DESC	...	3353
	EC	A3	C2	9E	0011B	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	...	3354
	F0	A3	01	DD	00121	PUSHL	#1	...	3355
		7E	A2	9A	00123	MOVZBL	AED_B_LINE, -(SP)	...	
	00000000G	00	02	FB	00127	CALLS	#2, SCRSET_CURSOR	...	
		EC	A3	9F	0012E	PUSHAB	ECHO_DESC	...	3356
			01	FB	00131	CALLS	#1, AED_PUTOUTPUT	...	
	0000G	CF	C2	3C	00136	MOVZWL	SEGMENT_SIZE, -(SP)	...	3357
		7E	6E	D6	0013B	INCL	(SP)	...	
			A2	9A	0013D	MOVZBL	AED_B_LINE, -(SP)	...	
	00000000G	00	02	FB	00141	CALLS	#2, SCRERASE_LINE	...	
20	A2	E8	01	81	00148	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	...	3358
		01	10	88	0014E	BISB2	#16, AED_L_FLAGS+1	...	3359
		7E	A2	9A	00152	MOVZBL	AED_B_COLUMN, -(SP)	...	3361
		7E	A2	9A	00156	MOVZBL	AED_B_LINE, -(SP)	...	
	0000G	CF	02	FB	0015A	CALLS	#2, AED_SET_CURSOR	...	
		01	A2	8F	AA	BICW2	#8200, AED_L_FLAGS+1	...	3363
			2008	94	00165	CLRB	TERM_CHAR	...	3364
			10	01	D0	MOVL	#1, R0	...	3365
					04	RET		...	3367
					04			...	

; Routine Size: 364 bytes, Routine Base: \$CODE\$ + 1F0E

ACT_FIND_STR - locate specific string

```
2940 3368 1 XSBTTL 'ACT_FIND_STR - locate specific string'
2941 3369 1 ROUTINE ACT_FIND_STR =
2942 3370 1
2943 3371 1 ++
2944 3372 1
2945 3373 1 FUNCTIONAL DESCRIPTION:
2946 3374 1
2947 3375 1 This routine obtains the string to be searched for. If the search
2948 3376 1 string is terminated by the ADVANCE action key, the search is in
2949 3377 1 the forward direction. If the search string is terminated by the
2950 3378 1 BACKUP action key, the search is in the backward direction. The
2951 3379 1 screen is scrolled as necessary to accomodate the next occurrence
2952 3380 1 of the search string.
2953 3381 1
2954 3382 1 CALLING SEQUENCE:
2955 3383 1 ACT_FIND_STR ()
2956 3384 1
2957 3385 1 INPUT PARAMETERS:
2958 3386 1 none
2959 3387 1
2960 3388 1 IMPLICIT INPUTS:
2961 3389 1 OWN storage
2962 3390 1
2963 3391 1 OUTPUT PARAMETERS:
2964 3392 1 none
2965 3393 1
2966 3394 1 IMPLICIT OUTPUTS:
2967 3395 1 none
2968 3396 1
2969 3397 1 ROUTINE VALUE:
2970 3398 1 1 if successful
2971 3399 1 error status otherwise
2972 3400 1
2973 3401 1 SIDE EFFECTS:
2974 3402 1 The line segment table is updated as necessary, ACE line pointers
2975 3403 1 are updated, and the object's ACL is updated as necessary.
2976 3404 1
2977 3405 1 --
2978 3406 1
2979 3407 2 BEGIN
2980 3408 2
2981 3409 2 LOCAL
2982 3410 2 STRING_INDEX : VECTOR [1,WORD]; ! Index into search string buffer
2983 3411 2
2984 3412 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2985 3413 2 SCR$SET CURSOR T2T, 1);
2986 3414 2 AED_PUTOUTPUT ($DESCRIPTOR ('Search string: '));
2987 3415 2 STRING_INDEX = 0;
2988 3416 2 WHILE T
2989 3417 2 DO
2990 3418 2 BEGIN
2991 3419 2 TERM_CHAR = AED_DECODEKEY ();
2992 3420 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
2993 3421 2 IF TERM_CHAR EQL 0 THEN RETURN 1;
2994 3422 2 IF AED_L_FLAGS[AED_V_ACTIONKEY]
2995 3423 2 OR TERM_CHAR EQL AED_C_CHAR_ESC
2996 3424 2 THEN
```

```
2997 3425 4 BEGIN
2998 3426 4 IF .TERM_CHAR EQL KEY_C_RUB_BOL
2999 3427 4 THEN
3000 3428 4 BEGIN
3001 3429 4 SCR$ERASE PAGE (21, 1);
3002 3430 4 SCR$SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3003 3431 4 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3004 3432 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3005 3433 4 TERM_CHAR = 0;
3006 3434 4 RETURN 1;
3007 3435 4 END
3008 3436 4 ELSE IF .TERM_CHAR EQL KEY_C_RUB_CHR
3009 3437 4 THEN
3010 3438 4 BEGIN
3011 3439 4 IF .STRING_INDEX GTR 0
3012 3440 4 THEN
3013 3441 4 BEGIN
3014 3442 4 STRING_INDEX = .STRING_INDEX - 1;
3015 3443 4 AED_PUTOUTPUT ($DESCRIPTOR (XCHAR (AED_C_CHAR_BS),
3016 3444 4 XCHAR (AED_C_CHAR_BS)));
3017 3445 4
3018 3446 4 END;
3019 3447 4 END
3020 3448 4 ELSE EXITLOOP;
3021 3449 4 END
3022 3450 4 ELSE IF .TERM_CHAR GEQ ' '
3023 3451 4 THEN
3024 3452 4 BEGIN
3025 3453 4 IF .TERM_CHAR GEQ 'a' AND .TERM_CHAR LEQ 'z'
3026 3454 4 THEN TERM_CHAR = .TERM_CHAR - 32; ! Convert lower to upper case
3027 3455 4 ECHO_DESC[DSC$W_LENGTH] = 1;
3028 3456 4 ECHO_DESC[DSC$A_POINTER] = TERM_CHAR;
3029 3457 4 AED_PUTOUTPUT (ECHO_DESC);
3030 3458 4 SEARCH_STRING[.STRING_INDEX] = .TERM_CHAR;
3031 3459 4 STRING_INDEX = .STRING_INDEX + 1;
3032 3460 4 END;
3033 3461 4 END;
3034 3462 4 SEARCH_SIZE = .STRING_INDEX;
3035 3463 4 SCR$ERASE PAGE (21, 1);
3036 3464 4 IF .SEARCH_SIZE EQL 0
3037 3465 4 THEN
3038 3466 4 BEGIN
3039 3467 4 SCR$SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3040 3468 4 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3041 3469 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3042 3470 4 TERM_CHAR = 0;
3043 3471 4 RETURN 1;
3044 3472 4 END;
3045 3473 4 IF .AED_L_FLAGS[AED_V_ACTIONKEY]
3046 3474 4 THEN
3047 3475 4 BEGIN
3048 3476 4 IF .TERM_CHAR EQL KEY_C_ADVANCE THEN AED_L_FLAGS[AED_V_BACKWARD] = 0;
3049 3477 4 IF .TERM_CHAR EQL KEY_C_BACKUP THEN AED_C_FLAGS[AED_V_BACKWARD] = 1;
3050 3478 4 END;
3051 3479 4 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3052 3480 4 TERM_CHAR = KEY_C_FIND_NXT;
3053 3481 4 RETURN 1;
```


: 3054
: 30553482 2
3483 1 END;

! End of routine ACT_FIND_STR

```
.PSECT $SPLITS,NOWRT,NOEXE,2

20 3A 67 6E 69 72 74 73 20 68 63 72 61 65 53 00000 P.AAB: .ASCII \Search string: \
                                0000F .BLKB 1
                                0000000F 00010 P.AAA: .LONG 15
                                00000000' 00014 .ADDRESS P.AAB
                                08 00018 P.AAD: .ASCII <8>
                                20 00019 .ASCII \
                                08 0001A .ASCII <8>
                                00000003 0001B .BLKB 1
                                00000000' 0001C P.AAC: .LONG 3
                                00000000' 00020 .ADDRESS P.AAD

.PSECT $CODE$,NOWRT,2

01FC 00000 ACT_FIND_STR:
58 0000G CF 9E 00002 .WORD Save R2,R3,R4,R5,R6,R7,R8
57 00000000G 00 9E 00007 MOVAB AED_PUTOUTPUT, R8
56 00000000G 00 9E 0000E MOVAB SCR$ERASE_PAGE, R7
55 0000' CF 9E 00015 MOVAB SCR$SET_CURSOR, R6
54 0000' CF 9E 0001A MOVAB AED_L_FLAGS, R5
01 A5 08 8A 0001F MOVAB TERM_CHAR, R4
01 DD 00023 BICB2 #8, AED_L_FLAGS+1
15 DD 00025 PUSHL #1
66 02 FB 00027 PUSHL #21
0000' CF 9F 0002A CALLS #2, SCR$SET_CURSOR
68 01 FB 0002E PUSHAB P.AAA
0000G CF 00 FB 00033 CALLS #1, AED_PUTOUTPUT
64 50 90 00038 CLRW STRING_INDEX
01 A5 08 8A 0003B 1$: CALLS #0, AED_DECODEKEY
52 64 9A 0003F MOVAB R0, TERM_CHAR
05 02 A5 7D 13 00042 BICB2 #8, AED_L_FLAGS+1
18 52 91 00049 MOVZBL TERM_CHAR, R2
23 52 91 0004E BEQL 8$
09 12 00051 BBS #5, AED_L_FLAGS+2, 2$
01 DD 00053 CMPB R2, #27
15 DD 00055 CMPB R2, #35
67 02 FB 00057 BNEQ 3$
29 52 11 0005A CMPB R2, #41
52 91 0005C BRB 7$
3D 12 0005F CMPB R2, #41
53 B5 00061 BNEQ 6$
CE 13 00063 TSTW STRING_INDEX
53 B7 00065 BEQL 1$
0000' CF 9F 00067 DECB STRING_INDEX
68 01 FB 0006B PUSHAB P.AAC
CALLS #1, AED_PUTOUTPUT
```

			C3	11	0006E	BRB	1\$	3436
	20		52	91	00070	4\$: CMPB	R2, #32	3450
			BE	1F	00073	BLSSU	1\$	
61	8F		52	91	00075	CMPB	R2, #97	3453
			09	1F	00079	BLSSU	5\$	
7A	8F		52	91	0007B	CMPB	R2, #122	
			03	1A	0007F	BGTRU	5\$	
	64		20	82	00081	SUBB2	#32, TERM_CHAR	3454
DC	A4		01	B0	00084	5\$: MOVW	#1, ECHO_DESC	3455
E0	A4		64	9E	00088	MOVAB	TERM_CHAR, ECHO_DESC+4	3456
		DC	A4	9F	0008C	PUSHAB	ECHO_DESC	3457
	68		01	FB	0008F	CALLS	#1, AED_PUTOUTPUT	
	50		53	3C	00092	MOVZWL	STRING_INDEX, R0	3458
08	A440		64	90	00095	MOVB	TERM_CHAR, SEARCH_STRING[R0]	
			53	B6	0009A	INCW	STRING_INDEX	3459
			95	11	0009C	BRB	1\$	3416
04	A4		53	B0	0009E	6\$: MOVW	STRING_INDEX, SEARCH_SIZE	3462
			01	DD	000A2	PUSHL	#1	3463
	67		15	DD	000A4	PUSHL	#21	
		04	02	FB	000A6	CALLS	#2, SCR\$ERASE_PAGE	
			A4	B5	000A9	TSTW	SEARCH_SIZE	3464
			15	12	000AC	BNEQ	9\$	
	7E	20	A5	9A	000AE	7\$: MOVZBL	AED_B_COLUMN, -(SP)	3467
	7E	24	A5	9A	000B2	MOVZBL	AED_B_LINE, -(SP)	
	66		02	FB	000B6	CALLS	#2, SCR\$SET_CURSOR	
01	A5	2008	8F	AA	000B9	BICW2	#8200, AED_C_FLAGS+1	3469
			64	94	000BF	CLRB	TERM_CHAR	3470
			1E	11	000C1	8\$: BRB	12\$	3471
12	02		05	E1	000C3	9\$: BBC	#5, AED_L_FLAGS+2, 11\$	3473
	0C		64	91	000C8	CMPB	TERM_CHAR, #12	3476
			04	12	000CB	BNEQ	10\$	
	01		01	8A	000CD	BICB2	#1, AED_L_FLAGS+1	
	0E		64	91	000D1	10\$: CMPB	TERM_CHAR, #14	3477
			04	12	000D4	BNEQ	11\$	
	01		01	88	000D6	BISB2	#1, AED_L_FLAGS+1	
	01		08	8A	000DA	11\$: BICB2	#8, AED_L_FLAGS+1	3479
	64		05	90	000DE	MOVB	#5, TERM_CHAR	3480
	50		01	D0	000E1	12\$: MOVL	#1, R0	3481
			04	000E4	RET			3483

; Routine Size: 229 bytes, Routine Base: \$CODE\$ + 207A

```
3057 3484 1 $SBTTL 'ACT_FIND_NXT - locate next occurrence of string'
3058 3485 1 ROUTINE ACT_FIND_NXT =
3059 3486 1
3060 3487 1 ++
3061 3488 1
3062 3489 1 FUNCTIONAL DESCRIPTION:
3063 3490 1
3064 3491 1 This routine searches for the next occurrence of the selected search
3065 3492 1 string. The direction of the search depends of the state of the
3066 3493 1 BACKWARD flag.
3067 3494 1
3068 3495 1 CALLING SEQUENCE:
3069 3496 1 ACT_FIND_NXT ()
3070 3497 1
3071 3498 1 INPUT PARAMETERS:
3072 3499 1 none
3073 3500 1
3074 3501 1 IMPLICIT INPUTS:
3075 3502 1 OWN storage
3076 3503 1
3077 3504 1 OUTPUT PARAMETERS:
3078 3505 1 none
3079 3506 1
3080 3507 1 IMPLICIT OUTPUTS:
3081 3508 1 none
3082 3509 1
3083 3510 1 ROUTINE VALUE:
3084 3511 1 1 if successful
3085 3512 1 error status otherwise
3086 3513 1
3087 3514 1 SIDE EFFECTS:
3088 3515 1 The line segment table is updated as necessary, ACE line pointers
3089 3516 1 are updated, and the object's ACL is updated as necessary.
3090 3517 1
3091 3518 1 --
3092 3519 1
3093 3520 1 BEGIN
3094 3521 1
3095 3522 1 LOCAL
3096 3523 1 START_SEGMENT : REF $BBLOCK, ! Line where search started
3097 3524 1 SEARCH_BEGIN : VECTOR [1,WORD], ! Where to start the search
3098 3525 1 SEARCH_END : VECTOR [1,WORD], ! Where the search ends
3099 3526 1 STRING_LOCATION, ! Location of the found string or 0
3100 3527 1 NEW_ACE ! String found in new ACE flag
3101 3528 1 MATCH_SEGMENT : REF $BBLOCK; ! Address of line that matched
3102 3529 1
3103 3530 1 IF .SEARCH_SIZE EQL 0
3104 3531 1 THEN
3105 3532 1 BEGIN
3106 3533 1 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3107 3534 1 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3108 3535 1 TERM_CHAR = 0;
3109 3536 1 RETURN 1;
3110 3537 1 END;
3111 3538 1 IF NOT .AED_L_FLAGS[AED_V_BACKWARD]
3112 3539 1 THEN
3113 3540 1 BEGIN
```

```
3114 3541 3
3115 3542 ! See if the specified string is within the current line.
3116 3543
3117 3544 SEARCH_BEGIN = .BUFFER_INDEX + .SEARCH_SIZE;
3118 3545 IF .SEARCH_BEGIN GEQ .SEGMENT_SIZE
3119 3546 OR .SEARCH_BEGIN + .SEARCH_SIZE GTR .SEGMENT_SIZE
3120 3547 THEN STRING_LOCATION = 0
3121 3548 ELSE STRING_LOCATION = CH$FIND_SUB (.SEGMENT_SIZE - .SEARCH_BEGIN,
3122 3549 INPUT_BUFFER[.SEARCH_BEGIN],
3123 3550 .SEARCH_SIZE, SEARCH_STRING);
3124 3551
3125 3552 IF .STRING_LOCATION NEQ 0
3126 3553 THEN
3127 3554 BEGIN
3128 3555 BUFFER_INDEX = .STRING_LOCATION - INPUT_BUFFER[0];
3129 3556 AED_B_COLUMN = .BUFFER_INDEX + 1;
3130 3557 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3131 3558 AED_L_FLAGS[AED_V_GOLDREY] = 0;
3132 3559 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3133 3560 TERM_CHAR = 0;
3134 3561 RETURN 1;
3135 3562 END;
3136 3563
3137 3564 ! The specified search string is not within the current line. Update the
3138 3565 text ACE with the current line segment. Then loop through the remaining
3139 3566 line segments in the line table looking for the search string. If it is
3140 3567 found beyond the end of the current ACE, update the ACL with the current
3141 3568 ACE. Otherwise, simply set the various pointers to point to the line
3142 3569 segment where the search string was found.
3143 3570
3144 3571 NEW_TEXT_LINE = AED_REPSEGMENT ();
3145 3572 START_SEGMENT = .NEW_TEXT_LINE;
3146 3573 AED_L_FLAGS[AED_V_ENDACL] = 0;
3147 3574 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
3148 3575 MATCH_SEGMENT = .NEW_TEXT_LINE[LINE_L_FLINK];
3149 3576 NEW_ACE = .AED_L_LASTLINE - EQL .NEW_TEXT_LINE;
3150 3577 SEARCH_BEGIN = 0;
3151 3578 UNTIL .MATCH_SEGMENT EQLA AED_Q_LINETABLE[LINE_L_FLINK]
3152 3579 DO
3153 3580 BEGIN
3154 3581 STRING_LOCATION = CH$FIND_SUB (.MATCH_SEGMENT[LINE_W_SIZE] -
3155 3582 .SEARCH_BEGIN,
3156 3583 VECTOR [MATCH_SEGMENT[LINE_T_TEXT],
3157 3584 .SEARCH_BEGIN, .BYTE],
3158 3585 .SEARCH_SIZE, SEARCH_STRING);
3159 3586
3160 3587 IF .STRING_LOCATION NEQ 0
3161 3588 THEN
3162 3589 BEGIN
3163 3590 IF .NEW_ACE
3164 3591 AND (.AED_L_FLAGS[AED_V_MODIFIED]
3165 3592 OR .AED_L_FLAGS[AED_V_INSERT]
3166 3593 OR .AED_L_FLAGS[AED_V_INSERTTEXT])
3167 3594 THEN
3168 3595 BEGIN
3169 3596 FINISH_ACE ();
3170 3597 IF .AED_L_FLAGS[AED_V_PROMPT]
3170 3598 AND .AED_L_FLAGS[AED_V_FIRSTCHAR]
3170 3599 THEN
```



```
3171 3598 7 BEGIN
3172 3599 7 NEW_TEXT_LINE[LINE V_DUMMY] = 1;
3173 3600 7 AED_W_TOTALSIZE = 0;
3174 3601 6 END;
3175 3602 6 AED_L_FLAGS[AED V_INSERTTEXT] = 0;
3176 3603 6 IF .AED_W_TOTALSIZE EQL 0
3177 3604 6 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
3178 3605 6 AED_COMPRESS T;
3179 3606 6 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
3180 3607 6 IF NOT .AED_L_STATUS
3181 3608 6 THEN
3182 3609 7 BEGIN
3183 3610 7 AED_L_FLAGS[AED V_ACERROR] = 1;
3184 3611 7 AED_POSITION (.AED_L_FIRSTLINE);
3185 3612 7 AED_COPSEGMENT (.AED_L_FIRSTLINE);
3186 3613 7 INSQUE (AED T_CURLINE[LINE_L_FLINK],
3187 3614 7 .AED_C_FIRSTLINE[LINE_L_BLINK]);
3188 3615 7 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
3189 3616 7 THEN AED_C_LASTLINE = AED T_CURLINE;
3190 3617 7 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
3191 3618 7 THEN AED_C_BEGINLINE = AED T_CURLINE;
3192 3619 7 AED_L_FIRSTLINE = AED T_CURLINE;
3193 3620 7 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
3194 3621 7 AND .AED_C_FLAGS[AED V_ENDACL]
3195 3622 7 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
3196 3623 7 BUFFER_INDEX = 0;
3197 3624 7 AED_B_COLUMN = 1;
3198 3625 7 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3199 3626 7 AED_L_FLAGS[AED V_GOLDREY] = 0;
3200 3627 7 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3201 3628 7 TERM_CHAR = 0;
3202 3629 7 RETURN 1;
3203 3630 6 END;
3204 3631 6 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
3205 3632 6 END;
3206 3633 6 AED_L_FIRSTLINE = AED_L_LASTLINE = .MATCH_SEGMENT;
3207 3634 6 AED_W_TOTALSIZE = .AED_C_FIRSTLINE[LINE_W_SIZE];
3208 3635 6 UNTIL .AED_L_FIRSTLINE[LINE_V_BEGINACE]
3209 3636 6 DO
3210 3637 7 BEGIN
3211 3638 7 AED_L_FIRSTLINE = .AED_L_FIRSTLINE[LINE_L_BLINK];
3212 3639 7 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_FIRSTLINE[LINE_W_SIZE];
3213 3640 7 END;
3214 3641 6 DO
3215 3642 7 BEGIN
3216 3643 7 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
3217 3644 7 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
3218 3645 7 END;
3219 3646 6 UNTIL .AED_L_LASTLINE[LINE_V_BEGINACE]
3220 3647 6 OR .AED_L_LASTLINE EQL .AED_C_LINETABLE[LINE_L_FLINK];
3221 3648 6 AED_W_TOTALSIZE = .AED_W_TOTALSIZE - .AED_L_LASTLINE[LINE_W_SIZE];
3222 3649 6 AED_L_LASTLINE = .AED_C_LASTLINE[LINE_L_BLINK];
3223 3650 6 AED_POSITION (.MATCH_SEGMENT);
3224 3651 6 AED_COPSEGMENT (.MATCH_SEGMENT);
3225 3652 6 INSQUE (AED T_CURLINE[LINE_L_FLINK],
3226 3653 6 .MATCH_SEGMENT[LINE_C_BLINK]);
3227 3654 6 IF .AED_L_BEGINLINE EQL .MATCH_SEGMENT
```

```
3228 3655 5 THEN AED L BEGINLINE = AED T CURLINE[LINE_L_FLINK];
3229 3656 5 IF .AED C FIRSTLINE EQL .MATCH_SEGMENT
3230 3657 5 THEN AED C FIRSTLINE = AED T CURLINE[LINE_L_FLINK];
3231 3658 5 IF .AED C LASTLINE EQL .MATCH_SEGMENT
3232 3659 5 THEN AED C LASTLINE = AED T CURLINE[LINE_L_FLINK];
3233 3660 5 AED L CURACE = .AED L FIRSTCINE[LINE_L_BINACE];
3234 3661 5 BUFFER_INDEX = .STRING_LOCATION - MATCH_SEGMENT[LINE_T_TEXT];
3235 3662 5 AED B COLUMN = .BUFFER_INDEX + 1;
3236 3663 5 AED SET CURSOR (.AED B LINE, .AED B COLUMN);
3237 3664 5 AED L FLAGS[AED V GO[KEY]] = 0;
3238 3665 5 AED L FLAGS[AED V ACTIONKEY] = 0;
3239 3666 5 TERM CHAR = 0;
3240 3667 5 RETURN 1;
3241 3668 5 END;
3242 3669 5 IF .AED L LASTLINE EQL .MATCH_SEGMENT THEN NEW_ACE = 1;
3243 3670 5 MATCH_SEGMENT = .MATCH_SEGMENT[LINE_L_FLINK];
3244 3671 5 SEARCH_BEGIN = 0;
3245 3672 5 END;
3246 3673 5 END
3247 3674 5 ELSE
3248 3675 5 BEGIN
3249 3676 5 NEW_TEXT_LINE = AED REPMSEGMENT ();
3250 3677 5 START_SEGMENT = .NEW_TEXT_LINE;
3251 3678 5 MATCH_SEGMENT = NEW_TEXT [INE[LINE_L_FLINK];
3252 3679 5 AED L FLAGS[AED V ENDACL] = 0;
3253 3680 5 AED L FLAGS[AED V INSERTTEXT] = 0;
3254 3681 5 NEW_ACE = 0;
3255 3682 5 SEARCH_END = .BUFFER_INDEX;
3256 3683 5 UNTIL .MATCH_SEGMENT EQLA AED Q LINETABLE[LINE_L_FLINK]
3257 3684 5 DO
3258 3685 5 BEGIN
3259 3686 5 STRING_LOCATION = CHSFIND_SUB (.SEARCH_END, MATCH_SEGMENT[LINE_T_TEXT],
3260 3687 5 .SEARCH_SIZE, SEARCH_STRING);
3261 3688 5 IF .STRING_LOCATION NEQ 0
3262 3689 5 THEN
3263 3690 5 BEGIN
3264 3691 5 IF .NEW_ACE
3265 3692 5 AND (.AED L FLAGS[AED V MODIFIED]
3266 3693 5 OR .AED L FLAGS[AED V INSERT]
3267 3694 5 OR .AED L FLAGS[AED V INSERTTEXT])
3268 3695 5 THEN
3269 3696 5 BEGIN
3270 3697 5 FINISH_ACE ();
3271 3698 5 IF .AED L FLAGS[AED V PROMPT]
3272 3699 5 AND .AED C FLAGS[AED V FIRSTCHAR]
3273 3700 5 THEN
3274 3701 5 BEGIN
3275 3702 5 NEW_TEXT_LINE[LINE V_DUMMY] = 1;
3276 3703 5 AED W TOTALSIZE = 0;
3277 3704 5 END;
3278 3705 5 IF .AED W TOTALSIZE EQL 0
3279 3706 5 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_FLINK];
3280 3707 5 AED COMPRESS T);
3281 3708 5 AED L STATUS = AED UPDATEACL (.AED W TOTALSIZE);
3282 3709 5 IF NOT .AED L STATUS
3283 3710 5 THEN
3284 3711 5 BEGIN
```

```
3285 3712 7 AED_L_FLAGS[AED_V_ACERROR] = 1;
3286 3713 7 AED_POSITION (.AED_L_FIRSTLINE);
3287 3714 7 AED_COPSEGMENT (.AED_L_FIRSTLINE);
3288 3715 7 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
3289 3716 7 .AED_L_FIRSTLINE[LINE_L_BLINK]);
3290 3717 7 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
3291 3718 7 THEN AED_L_LASTLINE = AED_T_CURLINE;
3292 3719 7 IF .AED_L_BEGINLINE EQL .AED_L_FIRSTLINE
3293 3720 7 THEN AED_L_BEGINLINE = AED_T_CURLINE;
3294 3721 7 AED_L_FIRSTLINE = AED_T_CURLINE;
3295 3722 7 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
3296 3723 7 AND .AED_L_FLAGS[AED_V_ENDACL]
3297 3724 7 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
3298 3725 7 BUFFER_INDEX = 0;
3299 3726 7 AED_B_COLUMN = 1;
3300 3727 7 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3301 3728 7 AED_L_FLAGS[AED_V_GOODKEY] = 0;
3302 3729 7 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3303 3730 7 TERM_CHAR = 0;
3304 3731 7 RETURN 1;
3305 3732 6 END;
3306 3733 6 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
3307 3734 5 END;
3308 3735 5 SEARCH_BEGIN = .SEARCH_END;
3309 3736 5 WHILE T
3310 3737 5 DO
3311 3738 6 BEGIN
3312 3739 6 SEARCH_BEGIN = .SEARCH_BEGIN - .SEARCH_SIZE;
3313 3740 6 STRING_LOCATION = CH$FIND_SUB (.SEARCH_END - .SEARCH_BEGIN,
3314 3741 6 VECTOR [MATCH_SEGMENT[LINE_T_TEXT],
3315 3742 6 .SEARCH_BEGIN, BYTE],
3316 3743 6 .SEARCH_SIZE, SEARCH_STRING);
3317 3744 6 IF .STRING_LOCATION NEQ 0
3318 3745 6 THEN
3319 3746 7 BEGIN
3320 3747 7 AED_L_FIRSTLINE = AED_L_LASTLINE = .MATCH_SEGMENT;
3321 3748 7 AED_W_TOTALSIZE = .AED_L_FIRSTLINE[LINE_W_SIZE];
3322 3749 7 UNTIL .AED_L_FIRSTLINE[LINE_V_BEGINACE]
3323 3750 7 DO
3324 3751 8 BEGIN
3325 3752 8 AED_L_FIRSTLINE = .AED_L_FIRSTLINE[LINE_L_BLINK];
3326 3753 8 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_FIRSTLINE[LINE_W_SIZE];
3327 3754 8 END;
3328 3755 7 DO
3329 3756 8 BEGIN
3330 3757 8 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
3331 3758 8 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
3332 3759 8 END;
3333 3760 7 UNTIL .AED_L_LASTLINE[LINE_V_BEGINACE]
3334 3761 7 OR .AED_L_LASTLINE EQL AED_Q_LINETABLE[LINE_L_FLINK];
3335 3762 7 AED_W_TOTALSIZE = .AED_W_TOTALSIZE - .AED_L_LASTLINE[LINE_W_SIZE];
3336 3763 7 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_BLINK];
3337 3764 7 AED_POSITION (.MATCH_SEGMENT);
3338 3765 7 AED_COPSEGMENT (.MATCH_SEGMENT);
3339 3766 7 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
3340 3767 7 .MATCH_SEGMENT[LINE_L_BLINK]);
3341 3768 7 IF .AED_L_BEGINLINE EQL .MATCH_SEGMENT
```



```
3342 3769 7 THEN AED_L_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
3343 3770 7 IF .AED_C_FIRSTLINE EQL .MATCH_SEGMENT
3344 3771 7 THEN AED_C_FIRSTLINE = AED_T_CURLINE[LINE_L_FLINK];
3345 3772 7 IF .AED_C_LASTLINE EQL .MATCH_SEGMENT
3346 3773 7 THEN AED_C_LASTLINE = AED_T_CURLINE[LINE_L_FLINK];
3347 3774 7 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
3348 3775 7 BUFFER_INDEX = .STRING_LOCATION - MATCH_SEGMENT[LINE_T_TEXT];
3349 3776 7 AED_B_COLUMN = BUFFER_INDEX + 1;
3350 3777 7 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3351 3778 7 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3352 3779 7 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3353 3780 7 TERM_CHAR = 0;
3354 3781 7 RETURN 1;
3355 3782 6 END;
3356 3783 5 END;
3357 3784 4 END;
3358 3785 4 IF .AED_L_FIRSTLINE EQL .MATCH_SEGMENT THEN NEW_ACE = 1;
3359 3786 4 MATCH_SEGMENT = .MATCH_SEGMENT[LINE_L_FLINK];
3360 3787 4 SEARCH_END = .MATCH_SEGMENT[LINE_W_SIZE];
3361 3788 4 END;
3362 3789 3 END;
3363 3790 2 SIGNAL (AED$ NOTFOUND);
3364 3791 2 AED_COPSEGMENT (.START_SEGMENT);
3365 3792 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .START_SEGMENT[LINE_L_FLINK]);
3366 3793 2 IF .AED_L_BEGINLINE EQL .START_SEGMENT THEN AED_L_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
3367 3794 2 IF .AED_L_FIRSTLINE EQL .START_SEGMENT THEN AED_L_FIRSTLINE = AED_T_CURLINE[LINE_L_FLINK];
3368 3795 2 IF .AED_L_LASTLINE EQL .START_SEGMENT THEN AED_C_LASTLINE = AED_T_CURLINE[LINE_L_FLINK];
3369 3796 2 AED_L_FLAGS[AED_V_ACERROR] = 1;
3370 3797 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3371 3798 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3372 3799 2 TERM_CHAR = 0;
3373 3800 2 RETURN 1;
3374 3801 2
3375 3802 1 END;
```

! End of routine ACT_FIND_NXT

OFFC 00000 ACT_FIND_NXT:						
	5B	0000'	CF 9E 00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	3485
	5A	0000'	CF 9E 00007	MOVAB	NEW_TEXT_LINE, R11	
	54	14	AB 3C 0000C	MOVAB	AED_L_FIRSTLINE, R10	
			03 12 00010	MOVZWL	SEARCH_SIZE, R4	3530
			0426 31 00012	BNEQ	1\$	
	03	C1	AA E9 00015	BRW	70\$	
			01D6 31 00019	BLBC	AED_L_FLAGS+1, 2\$	3538
57	E8	AB	54 A1 0001C	BRW	33\$	
			57 3C 00021	ADDW3	R4, BUFFER_INDEX, SEARCH_BEGIN	3544
			51 3C 00024	MOVZWL	SEARCH_BEGIN, R0	3545
		78	AA 3C 00028	MOVZWL	SEGMENT_SIZE, R1	
			50 D1 0002B	CMPL	R0, R1	
			09 18 0002B	BGEQ	3\$	
52	50		54 C1 0002D	ADDL3	R4, R0, R2	3546
	51		52 D1 00031	CMPL	R2, R1	
			04 15 00034	BLEQ	4\$	
			52 D4 00036	CLRL	STRING_LOCATION	3547

00B4 CA40	51	18	AB	51	17	11	00038	BRB	6\$			
				50	11	0003A	4\$:	SUBL2	R0, R1			3548
				54	39	0003D		MATCHC	R4, SEARCH_STRING, R1, INPUT_BUFFER[R0]			3549
				03	13	00046		BEQL	5\$			
				53	54	00048		MOVL	R4, R3			
				53	54	0004B	5\$:	SUBL2	R4, R3			
				52	53	0004E		MOVL	R3, STRING_LOCATION			
				0D	13	00051	6\$:	BEQL	7\$			3551
				50	CA	9E	00053	MOVAB	INPUT_BUFFER, R0			3554
	EB	AB		52	50	C3	00058	SUBL3	R0, STRING_LOCATION, BUFFER_INDEX			
					032A	31	0005D	BRW	59\$			3555
		0000G		CF	00	FB	00060	7\$:	CALLS	#0, AED_REPSEGMENT		3570
				6B	50	D0	00065	MOVL	R0, NEW_TEXT_LINE			
				51	6B	D0	00068	MOVL	NEW_TEXT_LINE, R1			3571
				56	51	D0	0006B	MOVL	R1, START_SEGMENT			
				AA	8F	AA	0006E	BICW2	#16416, AED_L_FLAGS			3573
		C0		54	61	D0	00074	MOVL	(R1), MATCH_SEGMENT			3574
					50	D4	00077	CLRL	R0			3575
				51	04	AA	D1	00079	CMPL	AED_L_LASTLINE, R1		
					02	12	0007D	BNEQ	8\$			
					50	D6	0007F	INCL	R0			
				58	50	D0	00081	8\$:	MOVL	R0, NEW_ACE		
					57	B4	00084	9\$:	CLRW	SEARCH_BEGIN		3576
				50	AA	9E	00086	MOVAB	AED_Q_INETABLE, R0			3577
				50	54	D1	0008A	CMPL	MATCH_SEGMENT, R0			
					03	12	0008D	BNEQ	10\$			
					0321	31	0008F	BRW	63\$			
				50	57	3C	00092	10\$:	MOVZWL	SEARCH_BEGIN, R0		3581
				51	A4	3C	00095	MOVZWL	8(MATCH_SEGMENT), R1			
				51	50	C2	00099	SUBL2	R0, R1			
				55	14	AB	3C	0009C	MOVZWL	SEARCH_SIZE, R5		3584
14 A044	51	18	AB	55	55	39	000A0	MATCHC	R5, SEARCH_STRING, R1, 20(R0)-[MATCH_SEGMENT]			3583
					03	13	000A8	BEQL	11\$			
				53	55	D0	000AA	MOVL	R5, R3			
				53	55	C2	000AD	11\$:	SUBL2	R5, R3		
				52	53	D0	000B0	MOVL	R3, STRING_LOCATION			
					03	12	000B3	BNEQ	12\$			3585
					012B	31	000B5	BRW	31\$			
				03	58	F8	000B8	12\$:	BLBS	NEW_ACE, 14\$		3588
					00A2	31	000BB	13\$:	BRW	23\$		
					AA	95	000BE	14\$:	TSTB	AED_L_FLAGS		3589
					0A	19	000C1	BLSS	15\$			
	05			AA	05	E0	000C3	BBS	#5, AED_L_FLAGS+1, 15\$			3590
	EE			AA	06	E1	000C8	BBC	#6, AED_L_FLAGS+1, 13\$			3591
		0000V		CF	00	FB	000CD	15\$:	CALLS	#0, FINISH_ACE		3594
					AA	95	000D2	TSTB	AED_L_FLAGS+1			3595
					10	18	000D5	BGEQ	16\$			
	OB			AA	04	E1	000D7	BBC	#4, AED_L_FLAGS+1, 16\$			3596
				50	6B	D0	000DC	MOVL	NEW_TEXT_LINE, R0			3599
				0A	04	88	000DF	BISB2	#4, 10(R0)			
					CA	B4	000E3	CLRW	AED_W_TOTALSIZE			3600
					8F	8A	000E7	16\$:	BICB2	#64, AED_L_FLAGS+1		3602
					CA	B5	000EC	TSTW	AED_W_TOTALSIZE			3603
					03	12	000F0	BNEQ	17\$			
					9B	D0	000F2	MOVL	NEW_TEXT_LINE, NEW_TEXT_LINE			3604
		0000G		CF	00	FB	000F5	17\$:	CALLS	#0, AED_COMPRESS		3605

0000G	7E	0284	CA	3C	000FA	MOVZWL	AED_W TOTALSIZE, -(SP)	3606
4C	CF		01	FB	000FF	CALLS	#1, AED_UPDATEACL	
	AA		50	DO	00104	MOVL	RO, AED_L STATUS	
CO	4E	4C	AA	E8	00108	BLBS	AED_L STATUS, 22\$	3607
	AA	40	8F	88	0010C	BISB2	#64, AED_L FLAGS	3610
0000G	CF		6A	DD	00111	PUSHL	AED_L FIRSTLINE	3611
			01	FB	00113	CALLS	#1, AED_POSITION	
0000G	CF		6A	DD	00118	PUSHL	AED_L FIRSTLINE	3612
			01	FB	0011A	CALLS	#1, AED_COPSEGMENT	
04	50		6A	DO	0011F	MOVL	AED_L FIRSTLINE, RO	3614
	BO	70	AA	OE	00122	INSQUE	AED_T_CURLINE, 24(RO)	
	6A	04	AA	D1	00127	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	3615
			05	12	00128	BNEQ	18\$	
04	AA	70	AA	9E	0012D	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	3616
	6A	08	AA	D1	00132	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	3617
			05	12	00136	BNEQ	19\$	
08	AA	70	AA	9E	00138	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	3618
	6A	70	AA	9E	0013D	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	3619
04	AA		6A	D1	00141	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	3620
			09	13	00145	BEQL	21\$	
04	CO		05	E1	00147	BBC	#5, AED_L FLAGS, 21\$	3621
	AA		20	8A	0014C	BICB2	#32, AED_L FLAGS	3622
		E8	AB	D4	00150	CLRL	BUFFER INDEX	3623
EO	AA		01	90	00153	MOVB	#1, AED_B_COLUMN	3624
		0236	31	00157	BRW	60\$		3625
CO	AA	2080	8F	AA	0015A	BICW2	#8320, AED_L FLAGS	3631
04	AA		54	DO	00160	MOVL	MATCH_SEGMENT, AED_L_LASTLINE	3633
	6A		54	DO	00164	MOVL	MATCH_SEGMENT, AED_L_FIRSTLINE	
	50		6A	DO	00167	MOVL	AED_L_FIRSTLINE, RO	3634
0284	CA	08	A0	B0	0016A	MOVW	8(RO), AED_W TOTALSIZE	
	50		6A	DO	00170	MOVL	AED_L_FIRSTLINE, RO	3635
	OF	0A	A0	E8	00173	BLBS	10(RO), 25\$	
	6A	04	A0	DO	00177	MOVL	4(RO), AED_L FIRSTLINE	3638
	50		6A	DO	0017B	MOVL	AED_L_FIRSTLINE, RO	3639
0284	CA	08	A0	A0	0017E	ADDW2	8(RO), AED_W TOTALSIZE	
			ED	11	00184	BRB	24\$	3635
	50	04	AA	DO	00186	MOVL	AED_L_LASTLINE, RO	3643
04	AA		60	DO	0018A	MOVL	(RO), AED_L_LASTLINE	
	50	04	AA	DO	0018E	MOVL	AED_L_LASTLINE, RO	3644
0284	CA	08	A0	A0	00192	ADDW2	8(RO), AED_W TOTALSIZE	
	09	0A	A0	E8	00198	BLBS	10(RO), 27\$	3646
	51	F0	AA	9E	0019C	MOVAB	AED_Q LINETABLE, R1	3647
	51		50	D1	001A0	CMPL	RO, RT	
			E5	12	001A3	BNEQ	26\$	
0284	CA	08	A0	A2	001A5	SUBW2	8(RO), AED_W TOTALSIZE	3648
04	AA	04	A0	DO	001AB	MOVL	4(RO), AED_L_LASTLINE	3649
			54	DO	001B0	PUSHL	MATCH_SEGMENT	3650
0000G	CF		01	FB	001B2	CALLS	#1, AED_POSITION	
			54	DD	001B7	PUSHL	MATCH_SEGMENT	3651
0000G	CF		01	FB	001B9	CALLS	#1, AED_COPSEGMENT	
04	B4	70	AA	OE	001BE	INSQUE	AED_T_CURLINE, 24(MATCH_SEGMENT)	3653
	54	08	AA	D1	001C3	CMPL	AED_L_BEGINLINE, MATCH_SEGMENT	3654
			05	12	001C7	BNEQ	28\$	
08	AA	70	AA	9E	001C9	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	3655
	54		6A	D1	001CE	CMPL	AED_L_FIRSTLINE, MATCH_SEGMENT	3656
			04	12	001D1	BNEQ	29\$	
	6A	70	AA	9E	001D3	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	3657

		54	04	AA	D1	001D7	29\$:	CMPL	AED_L_LASTLINE, MATCH_SEGMENT	3658
				03	13	001DB		BEQL	30\$	
				0199	31	001DD		BRW	58\$	
				0191	31	001E0	30\$:	BRW	57\$	
		54	04	AA	D1	001E3	31\$:	CMPL	AED_L_LASTLINE, MATCH_SEGMENT	3669
				03	12	001E7		BNEQ	32\$	
		58		01	D0	001E9		MOVL	#1, NEW ACE	
		54		64	D0	001EC	32\$:	MOVL	(MATCH_SEGMENT), MATCH_SEGMENT	3670
				FE92	31	001EF		BRW	9\$	3671
0000G	CF			00	FB	001F2	33\$:	CALLS	#0, AED_REPSEGMENT	3676
	6B			50	D0	001F7		MOVL	R0, NEW_TEXT_LINE	
	56			6B	D0	001FA		MOVL	NEW_TEXT_LINE, START_SEGMENT	3677
	54			6B	D0	001FD		MOVL	NEW_TEXT_LINE, MATCH_SEGMENT	3678
	CO	AA	4020	8F	AA	00200		BICW2	#16, AED_L_FLAGS	3680
				58	D4	00206		CLRL	NEW ACE	3681
	59			E8	AB	00208		MOVW	BUFFER_INDEX, SEARCH_END	3682
	50			F0	AA	9E 0020C	34\$:	MOVAB	AED_Q_CINETABLE, R0	3683
	50			54	D1	00210		CMPL	MATCH_SEGMENT, R0	
				03	12	00213		BNEQ	35\$	
				019B	31	00215		BRW	63\$	
14	A4	59	18	AB	3C	00218	35\$:	MOVZWL	SEARCH_SIZE, R5	3687
				55	39	0021C		MATCHC	R5, SEARCH_STRING, SEARCH_END, -	3686
									20(MATCH_SEGMENT)	
				03	13	00223		BEQL	36\$	
		53		55	D0	00225		MOVL	R5, R3	
		53		55	C2	00228	36\$:	SUBL2	R5, R3	
		52		53	D0	0022B		MOVL	R3, STRING_LOCATION	
				03	12	0022E		BNEQ	37\$	3688
				016D	31	00230		BRW	61\$	
		03		58	E8	00233	37\$:	BLBS	NEW_ACE, 39\$	3691
				0095	31	00236	38\$:	BRW	48\$	
				CO	AA	95 00239	39\$:	TSTB	AED_L_FLAGS	3692
					0A	19 0023C		BLSS	40\$	
05	C1	AA		05	E0	0023E		BBS	#5, AED_L_FLAGS+1, 40\$	3693
EE	C1	AA		06	E1	00243		BBC	#6, AED_L_FLAGS+1, 38\$	3694
	0000V	CF		00	FB	00248	40\$:	CALLS	#0, FINISH ACE	3697
				C1	AA	95 0024D		TSTB	AED_L_FLAGS+1	3698
					10	18 00250		BGEQ	41\$	
OB	C1	AA		04	E1	00252		BBC	#4, AED_L_FLAGS+1, 41\$	3699
		50		6B	D0	00257		MOVL	NEW_TEXT_LINE, R0	3702
	OA	AO		04	88	0025A		BISB2	#4, -10(R0)	
			0284	CA	B4	0025E		CLRW	AED_W_TOTALSIZE	3703
			0284	CA	B5	00262	41\$:	TSTW	AED_W_TOTALSIZE	3705
				03	12	00266		BNEQ	42\$	
	7B			9B	D0	00268		MOVL	NEW_TEXT_LINE, NEW_TEXT_LINE	3706
0000G	CF			00	FB	0026B	42\$:	CALLS	#0, AED_COMPRESS	3707
	7E		0284	CA	3C	00270		MOVZWL	AED_W_TOTALSIZE, -(SP)	3708
0000G	CF			01	FB	00275		CALLS	#1, AED_UPDATEACL	
	4C	AA		50	D0	0027A		MOVL	R0, AED_L_STATUS	
		46		4C	AA	E8 0027E		BLBS	AED_L_STATUS, 47\$	3709
	CO	AA		40	8F	88 00282		BISB2	#64, AED_L_FLAGS	3712
				6A	DD	00287		PUSHL	AED_L_FIRSTLINE	3713
0000G	CF			01	FB	00289		CALLS	#1, AED_POSITION	
				6A	DD	0028E		PUSHL	AED_L_FIRSTLINE	3714
0000G	CF			01	FB	00290		CALLS	#1, AED_COPSEGMENT	
	50			6A	D0	00295		MOVL	AED_L_FIRSTLINE, R0	3716
	04	BO		70	AA	0E 00298		INSQUE	AED_T_CURLINE, #4(R0)	

	6A	04	AA	D1	0029D	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	3717
			05	12	002A1	BNEQ	43\$	
04	AA	70	AA	9E	002A3	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	3718
	6A	08	AA	D1	002AB	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	3719
			05	12	002AC	BNEQ	44\$	
08	AA	70	AA	9E	002AE	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	3720
	6A	70	AA	9E	002B3	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	3721
04	AA		6A	D1	002B7	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	3722
			03	12	002BB	BNEQ	46\$	
			FE90	31	002BD	BRW	21\$	
F8	CO	AA	05	E1	002C0	BBC	#5, AED_L_FLAGS, 45\$	3723
			FE84	31	002C5	BRW	20\$	
	CO	AA	8F	AA	002C8	BICW2	#8320, AED_L_FLAGS	3733
	57	2080	59	B0	002CE	MOVW	SEARCH_END, SEARCH_BEGIN	3735
	57	14	AB	A2	002D1	SUBW2	SEARCH_SIZE, SEARCH_BEGIN	3739
	50		57	3C	002D5	MOVZWL	SEARCH_BEGIN, R0	3740
	51		59	3C	002D8	MOVZWL	SEARCH_END, R1	
	51		50	C2	002DB	SUBL2	R0, R1	
14 A044	51	18	AB	3C	002DE	MOVZWL	SEARCH_SIZE, R5	3743
			55	39	002E2	MATCHC	R5, SEARCH_STRING, R1, 20(R0)- [MATCH_SEGMENT]	3742
			03	13	002EA	BEQL	50\$	
	53		55	D0	002EC	MOVL	R5, R3	
	53		55	C2	002EF	SUBL2	R5, R3	
	52		53	D0	002F2	MOVL	R3, STRING_LOCATION	
			DA	13	002F5	BEQL	49\$	3744
04	AA		54	D0	002F7	MOVL	MATCH_SEGMENT, AED_L_LASTLINE	3747
	6A		54	D0	002FB	MOVL	MATCH_SEGMENT, AED_L_FIRSTLINE	
	50		6A	D0	002FE	MOVL	AED_L_FIRSTLINE, R0	3748
0284	CA	08	A0	B0	00301	MOVW	8(R0), AED_W_TOTALSIZE	
	50		6A	D0	00307	MOVL	AED_L_FIRSTLINE, R0	3749
	0F	0A	A0	E8	0030A	BLBS	10(R0), 52\$	
	6A	04	A0	D0	0030E	MOVL	4(R0), AED_L_FIRSTLINE	3752
	50		6A	D0	00312	MOVL	AED_L_FIRSTLINE, R0	3753
0284	CA	08	A0	A0	00315	ADDW2	8(R0), AED_W_TOTALSIZE	
			ED	11	0031B	BRB	51\$	3749
	50	04	AA	D0	0031D	MOVL	AED_L_LASTLINE, R0	3757
04	AA		60	D0	00321	MOVL	(R0), AED_L_LASTLINE	
	50	04	AA	D0	00325	MOVL	AED_L_LASTLINE, R0	3758
0284	CA	08	A0	A0	00329	ADDW2	8(R0), AED_W_TOTALSIZE	
	09	0A	A0	E8	0032F	BLBS	10(R0), 54\$	3760
	51	F0	AA	9E	00333	MOVAB	AED_Q_LINETABLE, R1	3761
	51		50	D1	00337	CMPL	R0, RT	
			E5	12	0033A	BNEQ	53\$	
0284	CA	08	A0	A2	0033C	SUBW2	8(R0), AED_W_TOTALSIZE	3762
04	AA	04	A0	D0	00342	MOVL	4(R0), AED_L_LASTLINE	3763
			54	DD	00347	PUSHL	MATCH_SEGMENT	3764
0000G	CF		01	FB	00349	CALLS	#1, AED_POSITION	
			54	DD	0034E	PUSHL	MATCH_SEGMENT	3765
0000G	CF		01	FB	00350	CALLS	#1, AED_COPSEGMENT	
04	B4	70	AA	0E	00355	INSQUE	AED_T_CURLINE, 34(MATCH_SEGMENT)	3767
	54	08	AA	D1	0035A	CMPL	AED_L_BEGINLINE, MATCH_SEGMENT	3768
			05	12	0035E	BNEQ	55\$	
08	AA	70	AA	9E	00360	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	3769
	54		6A	D1	00365	CMPL	AED_L_FIRSTLINE, MATCH_SEGMENT	3770
			04	12	00368	BNEQ	56\$	
	6A	70	AA	9E	0036A	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	3771

		54	04	AA	D1	0036E	568:	CMPL	AED_L_LASTLINE, MATCH_SEGMENT	3772	
				05	12	00372		BNEQ	588		
		04	AA	70	AA	9E	00374	578:	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	3773
		50		6A	DO	00379	588:	MOVL	AED_L_FIRSTLINE, R0	3774	
		FC	AA	0C	A0	DO	0037C		MOVL	12(R0), AED_L_CURACE	
53		52		54	C3	00381		SUBL3	MATCH_SEGMENT-STRING_LOCATION, R3	3775	
		E8	AB	EC	A3	9E	00385		MOVAB	-20(R3), BUFFER_INDEX	
EO	AA	E8	AB	01	81	0038A	598:	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	3776	
		7E		EO	AA	9A	00390	608:	MOVZBL	AED_B_COLUMN, -(SP)	3777
		7E		E4	AA	9A	00394		MOVZBL	AED_B_LINE, -(SP)	
		0000G	CF	02	FB	00398		CALLS	#2, AED_SET_CURSOR		
				009B	31	0039D		BRW	708	3778	
		54		6A	D1	003A0	618:	CMPL	AED_L_FIRSTLINE, MATCH_SEGMENT	3785	
				03	12	003A3		BNEQ	628		
		58		01	DO	003A5		MOVL	#1, NEW_ACE		
		54	04	A4	DO	003AB	628:	MOVL	4(MATCH_SEGMENT), MATCH_SEGMENT	3786	
		59	08	A4	DO	003AC		MOVW	8(MATCH_SEGMENT), SEARCH_END	3787	
				FE59	31	003B0		BRW	348	3683	
16	CO	AA		03	E1	003B3	638:	BBC	#3, AED_L_FLAGS, 648	3790	
				01	DD	003B8		PUSHL	#1		
				15	DD	003BA		PUSHL	#21		
	00000000G	00		02	FB	003BC		CALLS	#2, SCRSErase_PAGE		
				01	DD	003C3		PUSHL	#1		
				15	DD	003C5		PUSHL	#21		
	00000000G	00		02	FB	003C7		CALLS	#2, SCR\$SET_CURSOR		
			00000000G	8F	DD	003CE	648:	PUSHL	#AED\$NOTFOUND		
	00000000G	00		01	FB	003D4		CALLS	#1, LIB\$SIGNAL		
OF	CO	AA		03	E1	003DB		BBC	#3, AED_L_FLAGS, 658		
		7E	EO	AA	9A	003E0		MOVZBL	AED_B_COLUMN, -(SP)		
		7E	E4	AA	9A	003E4		MOVZBL	AED_B_LINE, -(SP)		
	00000000G	00		02	FB	003E8		CALLS	#2, SCR\$SET_CURSOR		
			00000000*	8F	D5	003EF	658:	TSTL	#<AED\$NOTFOUND&7>		
				14	13	003F5		BEQL	668		
00000000*	BF	D4	AA	03	00	ED	003F7	CMPTV	#0, #3, AED_L_WORSTERR, #<AED\$NOTFOUND&7>		
				08	18	00401		BGEQ	668		
	D4	AA	00000000G	8F	DO	00403		MOVL	#AED\$NOTFOUND, AED_L_WORSTERR		
				56	DD	0040B	668:	PUSHL	START_SEGMENT	3791	
	0000G	CF		01	FB	0040D		CALLS	#1, AED_COPY_SEGMENT		
	04	B6	70	AA	0E	00412		INSQUE	AED_T_CURLINE, 24(START_SEGMENT)	3792	
		56	08	AA	D1	00417		CMPL	AED_L_BEGINLINE, START_SEGMENT	3793	
				05	12	0041B		BNEQ	678		
	08	AA	70	AA	9E	0041D		MOVAB	AED_T_CURLINE, AED_L_BEGINLINE		
		56		6A	D1	00422	678:	CMPL	AED_L_FIRSTLINE, START_SEGMENT	3794	
				04	12	00425		BNEQ	688		
		6A	70	AA	9E	00427		MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE		
		56	04	AA	D1	0042B	688:	CMPL	AED_L_LASTLINE, START_SEGMENT	3795	
				05	12	0042F		BNEQ	698		
	04	AA	70	AA	9E	00431		MOVAB	AED_T_CURLINE, AED_L_LASTLINE		
	CO	AA	40	8F	88	00436	698:	BISB2	#64, AED_L_FLAGS	3796	
	C1	AA	2008	8F	AA	0043B	708:	BICW2	#8200, AED_L_FLAGS+1	3798	
			10	AB	94	00441		CLRB	TERM_CHAR	3799	
		50		01	DO	00444		MOVL	#1, R0	3800	
				04	00447			RET		3802	

; Routine Size: 1096 bytes, Routine Base: \$CODE\$ + 215F

```
3377 3803 1 XSBTTL 'ACT ADV FIELD - advance to the next field'
3378 3804 1 ROUTINE ACT_ADV_FIELD =
3379 3805 1
3380 3806 1 **
3381 3807 1
3382 3808 1 FUNCTIONAL DESCRIPTION:
3383 3809 1
3384 3810 1     This routine moves the cursor to the beginning of the next major
3385 3811 1     field or inserts the text for the first item in the next major
3386 3812 1     field depending on the state of the PROMPT flag. The cursor is
3387 3813 1     left positioned to the end of the selected field.
3388 3814 1
3389 3815 1 CALLING SEQUENCE:
3390 3816 1     ACT_ADV_FIELD ()
3391 3817 1
3392 3818 1 INPUT PARAMETERS:
3393 3819 1     none
3394 3820 1
3395 3821 1 IMPLICIT INPUTS:
3396 3822 1     OWN storage
3397 3823 1
3398 3824 1 OUTPUT PARAMETERS:
3399 3825 1     none
3400 3826 1
3401 3827 1 IMPLICIT OUTPUTS:
3402 3828 1     none
3403 3829 1
3404 3830 1 ROUTINE VALUE:
3405 3831 1     1 if successful
3406 3832 1     error status otherwise
3407 3833 1
3408 3834 1 SIDE EFFECTS:
3409 3835 1     The line segment table is updated as necessary, ACE line pointers
3410 3836 1     are updated, and the object's ACL is updated as necessary.
3411 3837 1
3412 3838 1 --
3413 3839 1
3414 3840 2 BEGIN
3415 3841 2
3416 3842 2 IF .AED_L_FLAGS[AED_V_OPENUIIC]
3417 3843 2 THEN
3418 3844 2     BEGIN
3419 3845 2         AED_L_FLAGS[AED_V_ACERROR] = 1;
3420 3846 2         SIGNAL (AED$ BADUIIC);
3421 3847 2         AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3422 3848 2         AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3423 3849 2         TERM CHAR = 0;
3424 3850 2         RETURN 1;
3425 3851 2     END
3426 3852 2 ELSE
3427 3853 2     BEGIN
3428 3854 2         IF NOT .AED_L_FLAGS[AED_V_NOITEMSEL]
3429 3855 2         THEN
3430 3856 2             BEGIN
3431 3857 2                 IF .BUFFER INDEX GTR 0
3432 3858 2                 AND (IF .AED_B_ACETYPE EQL ACESC_DIRDEF
3433 3859 2                     THEN .AED_B_FIELD LSS 5
```

ACT_ADV_FIELD - advance to the next field

```
3434 3860 5      ELSE .AED_B_FIELD LSS 6)
3435 3861 4      THEN
3436 3862 3      BEGIN
3437 3863 2      IF .INPUT_BUFFER[.BUFFER_INDEX - 1] EQL '+'
3438 3864 1      THEN INPUT_BUFFER[.BUFFER_INDEX - 1] = ','
3439 3865 0      ELSE
3440 3866 0      BEGIN
3441 3867 0      IF .BUFFER_INDEX GEQ .AED_L_PAGewidth
3442 3868 0      THEN AED_SEGSPLIT (BUFFER_INDEX, 0, 0, 0);
3443 3869 0      INPUT_BUFFER[.BUFFER_INDEX] = ',';
3444 3870 0      ECHO_DESC[DESC$W_LENGTH] = 1;
3445 3871 0      ECHO_DESC[DESC$A_POINTER] = INPUT_BUFFER[.BUFFER_INDEX];
3446 3872 0      AED_PUTOUTPUT (ECHO_DESC);
3447 3873 0      SEGMENT_SIZE = .SEGMENT_SIZE + 1;
3448 3874 0      BUFFER_INDEX = .BUFFER_INDEX + 1;
3449 3875 0      AED_B_COLUMN = .AED_B_COLUMN + 1;
3450 3876 0      END;
3451 3877 0      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3452 3878 0      TERM_CHAR = KEY_C_SEL_FIELD;
3453 3879 0      RETURN 1;
3454 3880 0      END;
3455 3881 0      END
3456 3882 0      ELSE
3457 3883 0      BEGIN
3458 3884 0      WHILE .BUFFER_INDEX LSS .SEGMENT_SIZE
3459 3885 0      DO
3460 3886 0      BEGIN
3461 3887 0      AED_SELECTFIELD (BUFFER_INDEX);
3462 3888 0      IF .INPUT_BUFFER[.BUFFER_INDEX] EQL ',' THEN EXITLOOP;
3463 3889 0      END;
3464 3890 0      AED_B_COLUMN = .BUFFER_INDEX + 1;
3465 3891 0      AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3466 3892 0      END;
3467 3893 0      END;
3468 3894 0      AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3469 3895 0      AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3470 3896 0      TERM_CHAR = 0;
3471 3897 0      RETURN 1;
3472 3898 0      END;
3473 3899 1      END;
```

! End of routine ACT_ADV_FIELD

		007C 0000 ACT_ADV_FIELD:			
	56	00000000G	BF	D0	00002
	55	00000000G	D0	9E	00009
	54	0000'	CF	9E	00010
	53	0000'	CF	9E	00015
	4D	02	A3	E9	0001A
	63	40	BF	88	0001E
12	63		03	E1	00022
			01	DD	00026
			15	DD	00028
	00000000G	00	02	FB	0002A
					.WORD
					Save R2,R3,R4,R5,R6
					#AED\$ BADUIC, R6
					SCR\$SET CURSOR, R5
					BUFFER_INDEX, R4
					AED_L_FLAGS, R3
					AED_L_FLAGS+2, 48
					#64, AED_L_FLAGS
					#3, AED_L_FLAGS, 18
					#1
					#21
					CALLS #2, SCR\$ERASE_PAGE

3804

3842
3845
3846

00000000*	BF	14	A3	03	00	01	DD	00031	PUSHL	#1			
					65	15	DD	00033	PUSHL	#21			
						02	FB	00035	CALLS	#2, SCR\$SET_CURSOR			
						56	DD	00038	PUSHL	R6			
	00000000G	00				01	FB	0003A	CALLS	#1, LIB\$SIGNAL			
		63				03	E1	00041	BBC	#3, AED_L_FLAGS, 2\$			
		7E			20	A3	9A	00045	MOVZBL	AED_B_COLUMN, -(SP)			
		7E			24	A3	9A	00049	MOVZBL	AED_B_LINE, -(SP)			
		65				02	FB	0004D	CALLS	#2, SCR\$SET_CURSOR			
					00000000*	8F	D5	00050	TSTL	#<AED\$_BADUIC&7>			
						10	13	00056	BEQL	3\$			
						00	ED	00058	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_BADUIC&7>			
						04	18	00062	BGEQ	3\$			
		14	A3			56	D0	00064	MOVL	R6, AED_L_WORSTERR			
						009F	31	00068	BRW	13\$		3847	
		6A	02	A3		03	E0	00068	BBS	#3, AED_L_FLAGS+2, 10\$		3854	
				50		64	D0	00070	MOVL	BUFFER_INDEX, R0		3857	
						F3	15	00073	BLEQ	3\$			
		09	00AB			C3	91	00075	CMPB	AED_B_ACETYPE, #9		3858	
						07	12	0007A	BNEQ	5\$			
		05	0090			C3	91	0007C	CMPB	AED_B_FIELD, #5		3859	
						05	11	00081	BRB	6\$			
		06	0090			C3	91	00083	CMPB	AED_B_FIELD, #6		3860	
						DE	1E	00088	BGEQU	3\$			
		2B	00C3	C340		91	0008A	CMPB	INPUT_BUFFER-1[R0], #43			3863	
						08	12	00090	BNEQ	7\$			
		00C3	C340			2C	90	00092	MOVB	#44, INPUT_BUFFER-1[R0]		3864	
						36	11	00098	BRB	9\$			
		18	A3			50	D1	0009A	CMPL	R0, AED_L_PAGEWIDTH		3867	
						0B	19	0009E	BLSS	8\$			
						7E	7C	000A0	CLRQ	-(SP)		3868	
						7E	D4	000A2	CLRL	-(SP)			
						54	DD	000A4	PUSHL	R4			
		0000G	CF			04	FB	000A6	CALLS	#4, AED_SEGSPLIT			
						00C4	C3	9E	000AB	MOVAB	INPUT_BUFFER, R0		3869
		00	B440			2C	90	000B0	MOVB	#44, \$BUFFER_INDEX[R0]			
		04	A4			01	B0	000B5	MOVW	#1, ECHO_DESC		3870	
		08	A4			00	B440	9E	000B9	MOVAB	\$BUFFER_INDEX[R0], ECHO_DESC+4		3871
						04	A4	9F	000BF	PUSHAB	ECHO_DESC		3872
		0000G	CF			01	FB	000C2	CALLS	#1, AED_PUTOUTPUT			
						00B8	C3	B6	000C7	INCW	SEGMENT_SIZE		3873
						64	D6	000CB	INCL	BUFFER_INDEX		3874	
						20	A3	96	000CD	INCB	AED_B_COLUMN		3875
		01	A3			08	8A	000D0	BICB2	#8, AED_L_FLAGS+1		3877	
		28	A4			08	90	000D4	MOVB	#8, TERM_CHAR		3878	
						39	11	000D8	BRB	14\$		3879	
						64	D0	000DA	MOVL	BUFFER_INDEX, R2		3884	
52		00B8	C3			00	ED	000DD	CMPZV	#0, #16, SEGMENT_SIZE, R2			
						12	15	000E4	BLEQ	12\$			
						54	DD	000E6	PUSHL	R4		3887	
		0000G	CF			01	FB	000E8	CALLS	#1, AED_SELECTFIELD			
						64	D0	000ED	MOVL	BUFFER_INDEX, R2		3888	
						2C	91	000F0	CMPB	INPUT_BUFFER[R2], #44			
						E5	12	000F6	BNEQ	11\$			
		20	A3			01	81	000F8	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN		3890	
						7E	A3	9A	000FD	MOVZBL	AED_B_COLUMN, -(SP)		3891
						7E	A3	9A	00101	MOVZBL	AED_B_LINE, -(SP)		

ACT_ADV_FIELD - advance to the next field

15-Sep-1984 23:47:16
14-Sep-1984 11:52:29

VAX-11 BLISS-32 V4.0-742
[ACLEDIT.SRC]AEDMAIN.B32;1

Page 127
(27)

0000G	CF		02	FB 00105		CALLS	#2, AED SET CURSOR
01	A3	2008	8F	AA 0010A	138:	BICW2	#B200, AED_C_FLAGS+1
		28	A4	94 00110		CLRB	TERM CHAR
50			01	DO 00113	148:	MOVL	#1, R0
				04 00116		RET	

3895
3896
3897
3899

; Routine Size: 279 bytes. Routine Base: \$CODE\$ + 25A7

AED
V04

.....

ACT_SEL_FIELD - select the next field

```
3475 3900 1 %SBTTL 'ACT_SEL_FIELD - select the next field'
3476 3901 1 ROUTINE ACT_SEL_FIELD =
3477 3902 1
3478 3903 1 ++
3479 3904 1
3480 3905 1 FUNCTIONAL DESCRIPTION:
3481 3906 1
3482 3907 1 This routine moves the cursor to the beginning of the next field or
3483 3908 1 inserts the text for the first item in the next field depending on
3484 3909 1 the state of the PROMPT flag. The cursor is left positioned to the
3485 3910 1 end of the selected field.
3486 3911 1
3487 3912 1 CALLING SEQUENCE:
3488 3913 1 ACT_SEL_FIELD ()
3489 3914 1
3490 3915 1 INPUT PARAMETERS:
3491 3916 1 none
3492 3917 1
3493 3918 1 IMPLICIT INPUTS:
3494 3919 1 OWN storage
3495 3920 1
3496 3921 1 OUTPUT PARAMETERS:
3497 3922 1 none
3498 3923 1
3499 3924 1 IMPLICIT OUTPUTS:
3500 3925 1 none
3501 3926 1
3502 3927 1 ROUTINE VALUE:
3503 3928 1 1 if successful
3504 3929 1 error status otherwise
3505 3930 1
3506 3931 1 SIDE EFFECTS:
3507 3932 1 The line segment table is updated as necessary, ACE line pointers
3508 3933 1 are updated, and the object's ACL is updated as necessary.
3509 3934 1
3510 3935 1 --
3511 3936 1
3512 3937 2 BEGIN
3513 3938 2
3514 3939 2 IF .AED_L_FLAGS[AED_V_OPENUIC]
3515 3940 2 THEN
3516 3941 2 BEGIN
3517 3942 2 AED_L_FLAGS[AED_V_ACERROR] = 1;
3518 3943 2 SIGNAL (AED$ BADUTC);
3519 3944 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3520 3945 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3521 3946 2 TERM CHAR = 0;
3522 3947 2 RETURN 1;
3523 3948 2 END
3524 3949 2 ELSE
3525 3950 2 BEGIN
3526 3951 2 IF .BUFFER_INDEX LSS .SEGMENT_SIZE
3527 3952 2 OR NOT .AED_L_FLAGS[AED_V_NOITEMSEL]
3528 3953 2 THEN
3529 3954 2 BEGIN
3530 3955 2 AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
3531 3956 2 AED_SELECTFIELD(BUFFER_INDEX);
```

ACT_SEL_FIELD - select the next field

```
3532 3957 4 IF NOT .AED_L_FLAGS[AED_V_NOITEMSEL]
3533 3958 THEN
3534 3959 BEGIN
3535 3960 ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
3536 3961 ECHO_DESC[DSC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
3537 3962 SCR$SET_CURSOR T.AED_B_LINE, 1;
3538 3963 AED_PUTOUTPUT (ECHO_DESC);
3539 3964 SCR$ERASE_LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
3540 3965 END;
3541 3966 AED_B_COLUMN = .BUFFER_INDEX + 1;
3542 3967 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
3543 3968 END;
3544 3969 END;
3545 3970 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3546 3971 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3547 3972 TERM_CHAR = 0;
3548 3973 RETURN 1;
3549 3974
3550 3975 1 END;

! End of routine ACT_SEL_FIELD
```

				003C 00000 ACT_SEL_FIELD:			
			55	00000000G	8F D0 00002	Save R2,R3,R4,R5	3901
			54	00000000G	00 9E 00009	MOVL #AED\$_BADUIC, R5	
			53	0000'	CF 9E 00010	MOVAB SCR\$SET_CURSOR, R4	
			52	0000'	CF 9E 00015	MOVAB BUFFER_INDEX, R3	
			4C	02	A2 E9 0001A	MOVAB AED_L_FLAGS, R2	
			62	40	BF 88 0001E	BLBC AED_L_FLAGS+2, 3\$	3939
12			62		03 E1 00022	BISB2 #64, AED_L_FLAGS	3942
					01 DD 00026	BBC #3, AED_L_FLAGS, 1\$	3943
					15 DD 00028	PUSHL #1	
	00000000G	00			02 FB 0002A	PUSHL #21	
					01 DD 00031	CALLS #2, SCR\$ERASE_PAGE	
					15 DD 00033	PUSHL #1	
			64		02 FB 00035	PUSHL #21	
					55 DD 00038	CALLS #2, SCR\$SET_CURSOR	
	00000000G	00			01 FB 0003A	PUSHL R5	
0B			62		03 E1 00041	CALLS #1, LIB\$SIGNAL	
			7E	20	A2 9A 00045	BBC #3, AED_L_FLAGS, 2\$	
			7E	24	A2 9A 00049	MOVZBL AED_B_COLUMN, -(SP)	
			64		02 FB 0004D	MOVZBL AED_B_LINE, -(SP)	
				00000000*	BF D5 00050	CALLS #2, SCR\$SET_CURSOR	
					71 13 00056	TSTL #<AED\$_BADUIC&7>	
00000000*	8F	14	A2	03	00 ED 00058	BEQ 6\$	
					65 18 00062	CMPZV #0, #3, AED_L_WORSTERR, #<AED\$_BADUIC&7>	
			14	A2	55 D0 00064	BGEQ 6\$	
					5F 11 00068	MOVL R5, AED_L_WORSTERR	
					00 ED 0006A	BRB 6\$	3944
63	00B8	C2	10		05 14 00071	CMPZV #0, #16, SEGMENT_SIZE, BUFFER_INDEX	3951
		51	02	A2	03 E0 00073	BGTB 4\$	
			01	A2	10 8A 00078	BBS #3, AED_L_FLAGS+2, 6\$	3952
					53 DD 0007C	BICB2 #16, AED_L_FLAGS+1	3953
					01 FB 0007E	PUSHL R3	3956
						CALLS #1, AED_SELECTFIELD	

ACT_SEL_FIELD - select the next field

M 3
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29VAX-11 B11ss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1Page 130
(28)

2F	02	A2	03	E0	00083	BBS	#3, AED_L_FLAGS+2, 5\$...	3957
	04	A3	C2	B0	00088	MOVW	AED_T_CURLINE+8, ECHO_DESC	...	3960
	08	A3	C2	9E	0008E	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	...	3961
			01	DD	00094	PUSHL	#1	...	3962
		7E	A2	9A	00096	MOVZBL	AED_B_LINE, -(SP)	...	
		64	02	FB	0009A	CALLS	#2, SCRSET_CURSOR	...	
			A3	9F	0009D	PUSHAB	ECHO_DESC	...	3963
	0000G	CF	01	FB	000A0	CALLS	#1, AED_PUTOUTPUT	...	
		7E	C2	3C	000A5	MOVZWL	SEGMENT_SIZE, -(SP)	...	3964
			6E	D6	000AA	INCL	(SP)	...	
		7E	A2	9A	000AC	MOVZBL	AED_B_LINE, -(SP)	...	
	00000000G	00	02	FB	000B0	CALLS	#2, SCRERASE_LINE	...	
20	A2	63	01	81	000B7	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	...	3966
		7E	A2	9A	000BC	MOVZBL	AED_B_COLUMN, -(SP)	...	3967
		7E	A2	9A	000C0	MOVZBL	AED_B_LINE, -(SP)	...	
	0000G	CF	02	FB	000C4	CALLS	#2, AED_SET_CURSOR	...	
	01	A2	8F	AA	000C9	BICW2	#8200, AED_L_FLAGS+1	...	3971
			A3	94	000CF	CLRB	TERM_CHAR	...	3972
		50	01	D0	000D2	MOVL	#1, R0	...	3973
			04	000D5	RET			...	3975

; Routine Size: 214 bytes, Routine Base: \$CODE\$ + 26BE

ACT_SEL_ITEM - select the next item

```
3552 3976 1 %SBTTL 'ACT_SEL_ITEM - select the next item'
3553 3977 1 ROUTINE ACT_SEL_ITEM =
3554 3978 1
3555 3979 1 **
3556 3980 1
3557 3981 1 FUNCTIONAL DESCRIPTION:
3558 3982 1
3559 3983 1 This routine selects the next item based upon the selected field.
3560 3984 1 The cursor is left positioned at the end of the selected item. This
3561 3985 1 is only valid for an ACE being entered in PROMPT mode.
3562 3986 1
3563 3987 1 CALLING SEQUENCE:
3564 3988 1 ACT_SEL_ITEM ()
3565 3989 1
3566 3990 1 INPUT PARAMETERS:
3567 3991 1 none
3568 3992 1
3569 3993 1 IMPLICIT INPUTS:
3570 3994 1 OWN storage
3571 3995 1
3572 3996 1 OUTPUT PARAMETERS:
3573 3997 1 none
3574 3998 1
3575 3999 1 IMPLICIT OUTPUTS:
3576 4000 1 none
3577 4001 1
3578 4002 1 ROUTINE VALUE:
3579 4003 1 1 if successful
3580 4004 1 error status otherwise
3581 4005 1
3582 4006 1 SIDE EFFECTS:
3583 4007 1 The line segment table is updated as necessary, ACE line pointers
3584 4008 1 are updated, and the object's ACL is updated as necessary.
3585 4009 1
3586 4010 1 --
3587 4011 1
3588 4012 2 BEGIN
3589 4013 2
3590 4014 2 ! Clear all key indicators in case an error is seen.
3591 4015 2
3592 4016 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3593 4017 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3594 4018 2 TERM_CHAR = 0;
3595 4019 2
3596 4020 2 ! Check to see if item selection is allowed.
3597 4021 2
3598 4022 2 IF NOT .AED_L_FLAGS[AED_V_PROMPT] OR NOT .AED_L_FLAGS[AED_V_INSERTTEXT]
3599 4023 2 THEN
3600 4024 2 BEGIN
3601 4025 2 AED_L_FLAGS[AED_V_ACERROR] = 1;
3602 4026 2 SIGRAE (AED$_NOTITEMSEL);
3603 4027 2 RETURN 1;
3604 4028 2 END;
3605 4029 2
3606 4030 2 IF .AED_L_FLAGS[AED_V_OPENUIC]
3607 4031 2 THEN
3608 4032 2 BEGIN
```

```

: 3609      4033      3      AED_L_FLAGS[AED_V_ACERROR] = 1;
: 3610      4034      3      SIGNAL (AED$_BADUI);
: 3611      4035      3      RETURN 1;
: 3612      4036      3      END;
: 3613      4037      3
: 3614      4038      3      ! No error conditions have been found, select the next item.
: 3615      4039      3
: 3616      4040      3      AED_L_FLAGS[AED_V_FIRSTCHAR] = 0;
: 3617      4041      3      AED_SELECTITEM (BUFFER_INDEX);
: 3618      4042      3      ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
: 3619      4043      3      ECHO_DESC[DSC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
: 3620      4044      3      SCR$SET_CURSOR T.AED_B_LINE, 1;
: 3621      4045      3      AED_PUTOUTPUT (ECHO_DESC);
: 3622      4046      3      SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
: 3623      4047      3      AED_B_COLUMN = .BUFFER_INDEX + 1;
: 3624      4048      3      AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
: 3625      4049      3
: 3626      4050      3      RETURN 1;
: 3627      4051      3
: 3628      4052      3      END;

```

! End of routine ACT_SEL_ITEM

				01FC 00000 ACT_SEL_ITEM:							
				58	00000000G	8F	D0	00002	.WORD	Save R2,R3,R4,R5,R6,R7,R8	3977
				57	00000000G	00	9E	00009	MOVL	#AED\$_BADUI, R8	
				56	00000000G	8F	D0	00010	MOVAB	LIB\$SIGNAL, R7	
				55	00000000G	00	9E	00017	MOVL	#AED\$_NOITEMSEL, R6	
				54	00000000G	00	9E	0001E	MOVAB	SCR\$ERASE_PAGE, R5	
				53	00000000G	00	9E	00023	MOVAB	BUFFER_INDEX, R4	
				52	00000000G	00	9E	0002A	MOVAB	SCR\$SET_CURSOR, R3	
				51	00000000G	00	9E	00029	MOVAB	AED_L_FLAGS, R2	
				50	00000000G	00	9E	00028	MOVAB	#B200, AED_L_FLAGS+1	4017
				49	00000000G	00	9E	00027	CLRB	TERM_CHAR	4018
				48	00000000G	00	9E	00026	TSTB	AED_C_FLAGS+1	4022
				47	00000000G	00	9E	00025	BGEQ	1\$	
				46	00000000G	00	9E	00024	BBS	#6, AED_L_FLAGS+1, 4\$	
				45	00000000G	00	9E	00023	BISB2	#64, AED_C_FLAGS	4025
				44	00000000G	00	9E	00022	BBC	#3, AED_C_FLAGS, 2\$	4026
				43	00000000G	00	9E	00021	PUSHL	#1	
				42	00000000G	00	9E	00020	PUSHL	#21	
				41	00000000G	00	9E	0001F	CALLS	#2, SCR\$ERASE_PAGE	
				40	00000000G	00	9E	0001E	PUSHL	#1	
				39	00000000G	00	9E	0001D	PUSHL	#21	
				38	00000000G	00	9E	0001C	CALLS	#2, SCR\$SET_CURSOR	
				37	00000000G	00	9E	0001B	PUSHL	R6	
				36	00000000G	00	9E	0001A	CALLS	#1, LIB\$SIGNAL	
				35	00000000G	00	9E	00019	BBC	#3, AED_L_FLAGS, 3\$	
				34	00000000G	00	9E	00018	MOVZBL	AED_B_COLUMN, -(SP)	
				33	00000000G	00	9E	00017	MOVZBL	AED_B_LINE, -(SP)	
				32	00000000G	00	9E	00016	CALLS	#2, SCR\$SET_CURSOR	
				31	00000000G	00	9E	00015	TSTL	#<AED\$_NOITEMSEL&7>	
				30	00000000G	00	9E	00014	BEQ	7\$	
				29	00000000G	00	9E	00013	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOITEMSEL&7>	
				28	00000000G	00	9E	00012	BGEQ	7\$	
				27	00000000G	00	9E	00011			
				26	00000000G	00	9E	00010			
				25	00000000G	00	9E	0000F			
				24	00000000G	00	9E	0000E			
				23	00000000G	00	9E	0000D			
				22	00000000G	00	9E	0000C			
				21	00000000G	00	9E	0000B			
				20	00000000G	00	9E	0000A			
				19	00000000G	00	9E	00009			
				18	00000000G	00	9E	00008			
				17	00000000G	00	9E	00007			
				16	00000000G	00	9E	00006			
				15	00000000G	00	9E	00005			
				14	00000000G	00	9E	00004			
				13	00000000G	00	9E	00003			
				12	00000000G	00	9E	00002			
				11	00000000G	00	9E	00001			
				10	00000000G	00	9E	00000			
				09	00000000G	00	9E	00000			
				08	00000000G	00	9E	00000			
				07	00000000G	00	9E	00000			
				06	00000000G	00	9E	00000			
				05	00000000G	00	9E	00000			
				04	00000000G	00	9E	00000			
				03	00000000G	00	9E	00000			
				02	00000000G	00	9E	00000			
				01	00000000G	00	9E	00000			
				00	00000000G	00	9E	00000			

	14	A2		56	D0	00080	MOVL	R6, AED_L_WORSTERR		
				46	11	00084	BRB	7\$		4027
		44	02	A2	E9	00086	BLBC	AED_L_FLAGS+2, 8\$		4030
		62	40	8F	88	0008A	BISB2	#64, AED_L_FLAGS		4033
OE		62		03	E1	0008E	BBC	#3, AED_C_FLAGS, 5\$		4034
				01	DD	00092	PUSHL	#1		
				15	DD	00094	PUSHL	#21		
		65		02	FB	00096	CALLS	#2, SCR\$ERASE_PAGE		
				01	DD	00099	PUSHL	#1		
				15	DD	0009B	PUSHL	#21		
		63		02	FB	0009D	CALLS	#2, SCR\$SET_CURSOR		
				58	DD	000A0	PUSHL	R8		
OB		67		01	FB	000A2	CALLS	#1, LIB\$SIGNAL		
		62		03	E1	000A5	BBC	#3, AED_L_FLAGS, 6\$		
		7E	20	A2	9A	000A9	MOVZBL	AED_B_COLUMN, -(SP)		
		7E	24	A2	9A	000AD	MOVZBL	AED_B_LINE, -(SP)		
		63		02	FB	000B1	CALLS	#2, SCR\$SET_CURSOR		
			00000000*	8F	D5	000B4	TSTL	#<AED\$_BADUIC&7>		
				5E	13	000BA	BEQL	9\$		
00000000*	8F	14	A2	03	00	ED	CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_BADUIC&7>		
				52	18	000C6	BGEQ	9\$		
		14	A2	58	D0	000C8	MOVL	R8, AED_L_WORSTERR		
				4C	11	000CC	BRB	9\$		4035
		01	A2	10	8A	000CE	BICB2	#16, AED_L_FLAGS+1		4040
				54	DD	000D2	PUSHL	R4		4041
		0000G	CF	01	FB	000D4	CALLS	#1, AED_SELECTITEM		
		04	A4	C2	B0	000D9	MOVW	AED_T_CORLINE+8, ECHO_DESC		4042
		08	A4	C2	9E	000DF	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4		4043
				01	DD	000E5	PUSHL	#1		4044
		7E	24	A2	9A	000E7	MOVZBL	AED_B_LINE, -(SP)		
		63		02	FB	000EB	CALLS	#2, SCR\$SET_CURSOR		
			04	A4	9F	000EE	PUSHAB	ECHO_DESC		4045
		0000G	CF	01	FB	000F1	CALLS	#1, AED_PUTOUTPUT		
		7E	00B8	C2	3C	000F6	MOVZWL	SEGMENT_SIZE, -(SP)		4046
				6E	D6	000FB	INCL	(SP)		
		7E	24	A2	9A	000FD	MOVZBL	AED_B_LINE, -(SP)		
		00000000G	00	02	FB	00101	CALLS	#2, SCR\$ERASE_LINE		
20	A2			01	81	00108	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN		4047
		64		A2	9A	0010D	MOVZBL	AED_B_COLUMN, -(SP)		4048
		7E	20	A2	9A	00111	MOVZBL	AED_B_LINE, -(SP)		
		7E	24	A2	9A	00115	CALLS	#2, AED_SET_CURSOR		
		0000G	CF	02	FB	0011A	MOVL	#1, R0		4050
				01	D0	0011D	RET			4052
				04	00	0011D				

; Routine Size: 286 bytes, Routine Base: \$CODE\$ + 2794

ACT_HELP - provide interactive help

```
3630 4053 1 %SBTTL 'ACT_HELP - provide interactive help'
3631 4054 1 ROUTINE ACT_HELP =
3632 4055 1
3633 4056 1 ++
3634 4057 1
3635 4058 1 FUNCTIONAL DESCRIPTION:
3636 4059 1
3637 4060 1     This routine supplies the interactive help to the user.
3638 4061 1
3639 4062 1 CALLING SEQUENCE:
3640 4063 1     ACT_HELP ()
3641 4064 1
3642 4065 1 INPUT PARAMETERS:
3643 4066 1     none
3644 4067 1
3645 4068 1 IMPLICIT INPUTS:
3646 4069 1     OWN storage
3647 4070 1
3648 4071 1 OUTPUT PARAMETERS:
3649 4072 1     none
3650 4073 1
3651 4074 1 IMPLICIT OUTPUTS:
3652 4075 1     none
3653 4076 1
3654 4077 1 ROUTINE VALUE:
3655 4078 1     1 if successful
3656 4079 1     error status otherwise
3657 4080 1
3658 4081 1 SIDE EFFECTS:
3659 4082 1     The line segment table is updated as necessary, ACE line pointers
3660 4083 1     are updated, and the object's ACL is updated as necessary.
3661 4084 1
3662 4085 1 --
3663 4086 1
3664 4087 2 BEGIN
3665 4088 2
3666 4089 2 SCR$SET_SCROLL (1, 24);
3667 4090 2 AED_GIVEHELP ();
3668 4091 2 ACT_REFRESH (0);
3669 4092 2 RETURN 1;
3670 4093 2
3671 4094 1 END;
```

! Refresh the screen

! End of routine ACT_HELP

0000 00000 ACT_HELP:						
				.WORD	Save nothing	4054
		18	DD 00002	PUSHL	#24	4089
		01	DD 00004	PUSHL	#1	
00000000G	00	02	FB 00006	CALLS	#2, SCR\$SET_SCROLL	
0000G	CF	00	FB 0000D	CALLS	#0, AED_GIVEHELP	4090
		7E	D4 00012	CLRL	-(SP)	4091
0000V	CF	01	FB 00014	CALLS	#1, ACT_REFRESH	
	50	01	D0 00019	MOVL	#1, R0	4092
		04	0001C	RET		4094

AEDSMAN
V04-000

ACT_HELP - provide interactive help

E 4
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 Bliss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 135
(30)

; Routine Size: 29 bytes, Routine Base: \$CODE\$ + 28B2

ACT_REFRESH - refresh the display

```
3673 4095 1 %SBTTL 'ACT_REFRESH - refresh the display'
3674 4096 1 ROUTINE ACT_REFRESH (RESET) =
3675 4097 1
3676 4098 1 ++
3677 4099 1
3678 4100 1 FUNCTIONAL DESCRIPTION:
3679 4101 1
3680 4102 1 This routine clears the screen and repaints the display to eliminate
3681 4103 1 any extraneous garbage that may have appeared on the screen.
3682 4104 1
3683 4105 1 CALLING SEQUENCE:
3684 4106 1 ACT_REFRESH (ARG1)
3685 4107 1
3686 4108 1 INPUT PARAMETERS:
3687 4109 1 ARG1: 1 = reinitialize display from object's original ACL
3688 4110 1 0 = reinitialize display from in core copy of object's ACL
3689 4111 1
3690 4112 1 IMPLICIT INPUTS:
3691 4113 1 OWN storage
3692 4114 1
3693 4115 1 OUTPUT PARAMETERS:
3694 4116 1 none
3695 4117 1
3696 4118 1 IMPLICIT OUTPUTS:
3697 4119 1 none
3698 4120 1
3699 4121 1 ROUTINE VALUE:
3700 4122 1 1 if successful
3701 4123 1 error status otherwise
3702 4124 1
3703 4125 1 SIDE EFFECTS:
3704 4126 1 The line segment table is updated as necessary, ACE line pointers
3705 4127 1 are updated, and the object's ACL is updated as necessary.
3706 4128 1
3707 4129 1 --
3708 4130 1
3709 4131 2 BEGIN
3710 4132 2
3711 4133 2 LOCAL
3712 4134 2 CURRENT_LINE : REF $BLOCK, ! Address of current line segment
3713 4135 2 ATR_ARGCIST : BLOCKVECTOR [2, ITMSS_ITEM, BYTE], ! ACL attribute descriptor
3714 4136 2 ACL_CONTEXT, ! ACL context for $CHANGE_ACL
3715 4137 2 ACE_POINTER : REF $BLOCK, ! Address of current ACE
3716 4138 2 ACE_NEWADDR : REF $BLOCK, ! Copy of current ACE
3717 4139 2 ACE_DESC : $BLOCK [DSC$C_S_BLN], ! Binary ACE descr
3718 4140 2 ACE_TEXT_DESC : $BLOCK [DSC$C_S_BLN], ! Text ACE descriptor
3719 4141 2 ACE_TEXT : $BLOCK [3072], ! ACE text storage
3720 4142 2 ACE_TEXT_SIZE, ! ACE text size
3721 4143 2 FIRST_CHAR, ! Addr of first char in segment
3722 4144 2 LAST_CHAR, ! Addr of last char in segment
3723 4145 2 NEW_TEXT_LINE : REF $BLOCK, ! Converted line storage addr
3724 4146 2 LINE_SEG_SIZE; ! Size of line segment
3725 4147 2
3726 4148 2 ! If this is a reset operation, deallocate all the in core ACL information
3727 4149 2 ! and rebuild it from the object's actual ACL.
3728 4150 2
3729 4151 2 IF .RESET
```

ACT_REFRESH - refresh the display

```
3730 4152 2 THEN
3731 4153 BEGIN
3732 4154 CHSFILL (0, 2*ITMS_ITEM, ATR_ARGLIST);
3733 4155
3734 4156 ! Go through the line segment table and deallocate all the old segments.
3735 4157 ! In addition, get rid of the copies of the binary ACEs.
3736 4158
3737 4159 UNTIL REMQUE (.AED_Q_LINETABLE[LINE_L_FLINK], CURRENT_LINE)
3738 4160 DO
3739 4161 BEGIN
3740 4162 IF .CURRENT_LINE[LINE_V_BEGINACE]
3741 4163 THEN IF .CURRENT_LINE[LINE_L_BINACE] NEQ 0
3742 4164 THEN DEALLOCATE (.$BLOCK[.CURRENT_LINE[LINE_L_BINACE], ACESB_SIZE]),
3743 4165 CURRENT_LINE[LINE_L_BINACE]);
3744 4166 DEALLOCATE (.CURRENT_LINE[LINE_W_SIZE] + $BYTEOFFSET (LINE_T_TEXT),
3745 4167 CURRENT_LINE);
3746 4168 END;
3747 4169
3748 4170 ! Allocate storage for the temporary ACL segment buffer.
3749 4171
3750 4172 AED_L_STATUS = ALLOCATE (512, AED_A_ACLBUFFER);
3751 4173 IF NOT .AED_L_STATUS
3752 4174 THEN
3753 4175 BEGIN
3754 4176 SIGNAL (.AED_L_STATUS);
3755 4177 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3756 4178 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3757 4179 TERM CHAR = 0;
3758 4180 RETURN .AED_L_WORSTERR OR STSM_INHIB_MSG;
3759 4181 END;
3760 4182
3761 4183 ! Re-read any ACL associated with the object.
3762 4184
3763 4185 ACL_CONTEXT = 0;
3764 4186 ATR_ARGLIST[0, ITMSW_ITMCD] = ACLSC_READACL;
3765 4187 ATR_ARGLIST[0, ITMSW_BUFSIZ] = 512;
3766 4188 ATR_ARGLIST[0, ITMSL_BUFADR] = .AED_A_ACLBUFFER;
3767 4189 WHILE 1
3768 4190 DO
3769 4191 BEGIN
3770 4192 AED_L_STATUS = $CHANGE_ACL (CHAN = .AED_W_OBJCHAN,
3771 4193 OBJTYP = AED_C_OBJTYP,
3772 4194 OBJNAM = AED_Q_OBJNAM,
3773 4195 ITMLST = ATR_ARGLIST,
3774 4196 CONXT = ACL_CONTEXT);
3775 4197
3776 4198 IF NOT .AED_L_STATUS
3777 4199 THEN
3778 4200 BEGIN
3779 4201 IF .AED_L_STATUS EQL SS$ACLEMPY
3780 4202 OR .AED_L_STATUS EQL SS$NOMOREACE
3781 4203 THEN EXIT[COOP];
3782 4204 SIGNAL (AED$READERR, 1, AED_Q_OBJNAM, .AED_L_STATUS, 0);
3783 4205 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3784 4206 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3785 4207 TERM CHAR = 0;
3786 4208 RETURN .AED_L_WORSTERR OR STSM_INHIB_MSG;
3786 4209 END;
```

```
3787 4209 4 ACE_POINTER = .AED_A_ACLBUFFER;  
3788 4210 4 UNTIL .ACE_POINTER GEQA .AED_A_ACLBUFFER + 512  
3789 4211 4 DO  
3790 4212 4 BEGIN  
3791 4213 4 IF .ACE_POINTER[ACESB_SIZE] EQL 0 THEN EXITLOOP;  
3792 4214 4 AED_L_STATUS = ALLOCATE (.ACE_POINTER[ACESB_SIZE], ACE_NEWADDR);  
3793 4215 4 IF NOT .AED_L_STATUS  
3794 4216 4 THEN  
3795 4217 4 BEGIN  
3796 4218 4 SIGNAL (.AED_L_STATUS);  
3797 4219 4 AED_L_FLAGS[AED_V_GOLDKEY] = 0;  
3798 4220 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;  
3799 4221 4 TERM_CHAR = 0;  
3800 4222 4 RETURN .AED_L_WORSTERR OR STSM_INHIB_MSG;  
3801 4223 4 END;  
3802 4224 4 CHSMOVE (.ACE_POINTER[ACESB_SIZE], .ACE_POINTER, .ACE_NEWADDR);  
3803 4225 4 ACE_DESC[DSCSA_POINTER] = .ACE_POINTER;  
3804 4226 4 ACE_DESC[DSCSW_LENGTH] = .ACE_POINTER[ACESB_SIZE];  
3805 4227 4 ACE_TEXT_DESC[DSCSA_POINTER] = ACE_TEXT;  
3806 4228 4 ACE_TEXT_DESC[DSCSW_LENGTH] = 3072;  
3807 4229 4 AED_L_STATUS = $FORMAT_ACL (ACLEN = ACE_DESC,  
3808 4230 4 ACLEN = ACE_TEXT_DESC,  
3809 4231 4 ACLSTR = ACE_TEXT_DESC,  
3810 4232 4 WIDTH = AED [PAGEWIDTH,  
3811 4233 4 TRMDSC = $DESCRIPTOR (0),  
3812 4234 4 INDENT = 0);  
3813 4235 4 ACE_TEXT_SIZE = .ACE_TEXT_DESC[DSCSW_LENGTH];  
3814 4236 4 FIRST_CHAR = ACE_TEXT;  
3815 4237 4 AED_L_FIRSTLINE = AED_L_LASTLINE = 0;  
3816 4238 4 WHILE (LAST_CHAR = CH$FIND_CH (.ACE_TEXT_SIZE, .FIRST_CHAR, 0)) GTR 0  
3817 4239 4 DO  
3818 4240 4 BEGIN  
3819 4241 4 LINE_SEG_SIZE = .LAST_CHAR - .FIRST_CHAR;  
3820 4242 4 AED_L_STATUS = ALLOCATE (.LINE_SEG_SIZE + $BYTEOFFSET (LINE_T_TEXT),  
3821 4243 4 NEW_TEXT_LINE);  
3822 4244 4 IF NOT .AED_L_STATUS  
3823 4245 4 THEN  
3824 4246 4 BEGIN  
3825 4247 4 SIGNAL (.AED_L_STATUS);  
3826 4248 4 AED_L_FLAGS[AED_V_GOLDKEY] = 0;  
3827 4249 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;  
3828 4250 4 TERM_CHAR = 0;  
3829 4251 4 RETURN .AED_L_WORSTERR OR STSM_INHIB_MSG;  
3830 4252 4 END;  
3831 4253 4 NEW_TEXT_LINE[LINE_W_SIZE] = .LINE_SEG_SIZE;  
3832 4254 4 NEW_TEXT_LINE[LINE_L_BINACE] = .ACE_NEWADDR;  
3833 4255 4 CHSMOVE T.ACE_TEXT_SIZE, .FIRST_CHAR, NEW_TEXT_LINE[LINE_T_TEXT]);  
3834 4256 4 INSQUE (.NEW_TEXT_LINE, .AED_Q_INETABLE[LINE [BLINK]]);  
3835 4257 4 IF .AED_L_FIRSTLINE EQL 0 THEN AED_L_FIRSTLINE = .NEW_TEXT_LINE;  
3836 4258 4 AED_L_LASTLINE = .NEW_TEXT_LINE;  
3837 4259 4 FIRST_CHAR = .LAST_CHAR + 1;  
3838 4260 4 ACE_TEXT_SIZE = .ACE_TEXT_SIZE - .LINE_SEG_SIZE - 1;  
3839 4261 4 END;  
3840 4262 4 IF .ACE_TEXT_SIZE GTR 0  
3841 4263 4 THEN  
3842 4264 4 BEGIN  
3843 4265 4 AED_L_STATUS = ALLOCATE (.ACE_TEXT_SIZE + $BYTEOFFSET (LINE_T_TEXT),
```



```
3844 4266 6 NEW_TEXT_LINE);
3845 4267 6 IF NOT .AED_L_STATUS
3846 4268 6 THEN
3847 4269 6 BEGIN
3848 4270 6 SIGNAL (.AED_L_STATUS);
3849 4271 6 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3850 4272 6 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3851 4273 6 TERM_CHAR = 0;
3852 4274 6 RETURN .AED_L_WORSTERR OR STSM_INHIB_MSG;
3853 4275 6 END;
3854 4276 6 NEW_TEXT_LINE[LINE_W_SIZE] = .ACE_TEXT_SIZE;
3855 4277 6 NEW_TEXT_LINE[LINE_L_BINACE] = .ACE_NEWADDR;
3856 4278 6 CHSMOVE T.ACE_TEXT_SIZE, FIRST_CHAR, NEW_TEXT_LINE[LINE_T_TEXT]);
3857 4279 6 INSQUE (.NEW_TEXT_LINE, .AED_Q_LINETABLE[LINE_L_BLINK]);
3858 4280 6 IF .AED_L_FIRSTLINE EQL 0 THEN .AED_L_FIRSTLINE = .NEW_TEXT_LINE;
3859 4281 6 AED_L_LASTLINE = .NEW_TEXT_LINE;
3860 4282 6 END;
3861 4283 6 AED_L_FIRSTLINE[LINE_V_BEGINACE] = 1;
3862 4284 6 IF .ACE_POINTER[ACESB_HIDDEN]
3863 4285 6 OR (.ACE_POINTER[ACESB_TYPE] NEQ ACESC_KEYID
3864 4286 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_BIJNL
3865 4287 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_AIJNL
3866 4288 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_ATJNL
3867 4289 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_AUDIT
3868 4290 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_ALARM
3869 4291 6 AND .ACE_POINTER[ACESB_TYPE] NEQ ACESC_DIRDEF)
3870 4292 6 THEN AED_L_FIRSTLINE[LINE_V_NOTOUCH] = 1;
3871 4293 6 AED_L_LAST[LINE_V_ENDACE] = 1;
3872 4294 6 ACE_POINTER = .ACE_POINTER + .ACE_POINTER[ACESB_SIZE];
3873 4295 6 END;
3874 4296 6 END;
3875 4297 6 DEALLOCATE (512, AED_A_ACLBUFFER);
3876 4298 6
3877 4299 6 ! If there is no ACL (the display is empty), set up to append the text
3878 4300 6 ! entered. Otherwise, set up to modify the first segment of the display.
3879 4301 6
3880 4302 6 IF .AED_Q_LINETABLE[LINE_L_FLINK] EQLA AED_Q_LINETABLE[LINE_L_FLINK]
3881 4303 6 THEN
3882 4304 6 BEGIN
3883 4305 6 AED_L_FLAGS[AED_V_ENDACL] = 1; ! At the end of the ACL
3884 4306 6 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
3885 4307 6 AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
3886 4308 6 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_Q_LINETABLE[LINE_L_BLINK]);
3887 4309 6 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
3888 4310 6 AED_L_FIRSTLINE[LINE_Q_FLAGS] = LINE_M_BEGINACE;
3889 4311 6 AED_L_CURACE = 0;
3890 4312 6 IF .AED_L_FLAGS[AED_V_PROMPT]
3891 4313 6 THEN
3892 4314 6 BEGIN
3893 4315 6 AED_B_ACETYPE = 0;
3894 4316 6 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
3895 4317 6 AED_SELECTFIELD (BUFFER_INDEX);
3896 4318 6 AED_B_COLUMN = .BUFFER_INDEX + 1;
3897 4319 6 END;
3898 4320 6 END
3899 4321 6 ELSE
3900 4322 6 BEGIN
```

```
3901 4323 4 AED COPSEGMENT (.AED_Q LINETABLE[LINE_L FLINK]);
3902 4324 4 INSQUE (AED_T CURLINE[LINE_L FLINK], AED_Q LINETABLE[LINE_L FLINK]);
3903 4325 4 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
3904 4326 4 AED_W_TOTALSIZE = .AED_C_FIRSTLINE[LINE_Q_SIZE];
3905 4327 4 UNTIL .AED_L_LASTLINE[LINE_V_ENDACE]
3906 4328 4 DO
3907 4329 4 BEGIN
3908 4330 4 IF .AED_L_LASTLINE EQLA AED_T_CURLINE
3909 4331 4 THEN AED_C_LASTLINE = .AED_C_LASTLINE[LINE_L FLINK];
3910 4332 4 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L FLINK];
3911 4333 4 AED_W_TOTALSIZE = .AED_Q_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
3912 4334 4 END;
3913 4335 4 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
3914 4336 4 IF .AED_L_FLAGS[AED_V_PROMPT]
3915 4337 4 THEN
3916 4338 4 BEGIN
3917 4339 4 AED_L_FLAGS[AED_V_NOITEMSEL] = 1;
3918 4340 4 AED_SELECTFIELD (BUFFER_INDEX);
3919 4341 4 AED_B_COLUMN = .BUFFER_INDEX + 1;
3920 4342 4 END;
3921 4343 4 END;
3922 4344 4 AED_L_BEGINLINE = .AED_Q LINETABLE[LINE_L FLINK];
3923 4345 4 AED_B_SAVE_LIN = 1;
3924 4346 4 AED_B_SAVE_COL = .AED_B_COLUMN;
3925 4347 4 END;
3926 4348 4
3927 4349 4 ! Now repaint the display.
3928 4350 4
3929 4351 4 SCR$ERASE PAGE (1, 1);
3930 4352 4 SCR$SET_SCROLL (1, 20);
3931 4353 4 IF .AED_L_FLAGS[AED_V_VT5X] OR .AED_L_FLAGS[AED_V_VT1XX] ! Set up the scrolling region
3932 4354 4 THEN AED_PUTOUTPUT T$DESCRIPTOR ('XCHAR(AED_C_CHAR_ESC), '='));
3933 4355 4 TEMP_LINE = 1;
3934 4356 4 NEW_TEXT_LINE = .AED_L_BEGINLINE;
3935 4357 4 DO
3936 4358 4 BEGIN
3937 4359 4 SCR$SET_CURSOR (.TEMP_LINE, 1);
3938 4360 4 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
3939 4361 4 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
3940 4362 4 AED_PUTOUTPUT (ECHO_DESC);
3941 4363 4 TEMP_LINE = .TEMP_LINE + 1;
3942 4364 4 IF .NEW_TEXT_LINE[LINE_V_REPLACE]
3943 4365 4 THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L FLINK];
3944 4366 4 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L FLINK];
3945 4367 4 END
3946 4368 4 UNTIL (.TEMP_LINE GTR 20)
3947 4369 4 OR (.NEW_TEXT_LINE EQLA AED_Q LINETABLE[LINE_L FLINK]);
3948 4370 4 SCR$SET_CURSOR (.AED_B_SAVE_LIN, .AED_B_SAVE_COL);
3949 4371 4 AED_L_FLAGS[AED_V_GO$DREY] = 0;
3950 4372 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
3951 4373 4 TERM_CHAR = 0;
3952 4374 4 RETURN 1;
3953 4375 4
3954 4376 4 END;

! End of routine ACT_REFRESH
```

.PSECT \$SPLITS,NOWRT,NOEXE,2

```
00 00024 P.AAF: .BYTE 0
00025 .BLKB 3
00000001 00028 P.AAE: .LONG 1
00000000' 0002C .ADDRESS P.AAF
18 00030 P.AAH: .ASCII <27>
3D 00031 .ASCII \=\\
00000002 00032 .BLKB 2
00000000' 00034 P.AAG: .LONG 2
00000000' 00038 .ADDRESS P.AAH
```

.EXTRN SYSSCHANGE_ACL, SYSSFORMAT_ACL

.PSECT \$CODE\$,NOWRT,2

OFFC 00000 ACT_REFRESH:

		5E	F3C4	CE	9E	00002	.WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	4096	
		03	04	AC	EB	00007	MOVAB	-3132(SP), SP		
				04E7	31	0000B	BLBS	RESET, 1\$	4151	
18	00	6E		00	2C	0000E	BRW	55\$		
			E8	AD		00013	MOVCS	#0, (SP), #0, #24, ATR_ARGLIST	4154	
	04	AE	0000'	DF	0F	00015	2\$:	REMQUE	0AED_Q_LINETABLE, CURRENT_LINE	4159
				37	1D	0001B	BVS	4\$		
		52	04	AE	D0	0001D	MOVL	CURRENT_LINE, R2	4162	
		17	0A	A2	E9	00021	BLBC	10(R2), -3\$		
			0C	A2	D5	00025	TSTL	12(R2)	4163	
				12	13	00028	BEQL	3\$		
			0C	A2	9F	0002A	PUSHAB	12(R2)	4165	
	04	AE	0C	B2	D0	0002D	MOVL	012(R2), 4(SP)		
			04	AE	9F	00032	PUSHAB	4(SP)		
	00000000G	00		02	FB	00035	CALLS	#2, LIB\$FREE_VM		
			04	AE	9F	0003C	3\$:	PUSHAB	CURRENT_LINE	4167
	04	AE	08	A2	3C	0003F	MOVZWL	8(R2), 4(SP)		
	04	AE		14	C0	00044	ADDL2	#20, 4(SP)		
			04	AE	9F	00048	PUSHAB	4(SP)		
	00000000G	00		02	FB	0004B	CALLS	#2, LIB\$FREE_VM		
				C1	11	00052	BRB	2\$	4159	
			0000'	CF	9F	00054	4\$:	PUSHAB	AED_A_ACLBUFFER	4172
	04	AE	0200	8F	3C	00058	MOVZWL	#512, -4(SP)		
			04	AE	9F	0005E	PUSHAB	4(SP)		
	00000000G	00		02	FB	00061	CALLS	#2, LIB\$GET_VM		
		56		50	D0	00068	MOVL	R0, VM STATUS		
		0A		56	E9	0006B	BLBC	VM STATUS, 5\$		
0200	8F	00		00	2C	0006E	MOVCS	#0, (SP), #0, #512, 0AED_A_ACLBUFFER		
			0000'	DF		00075				
		0000'	CF	56	D0	00078	5\$:	MOVL	VM STATUS, AED_L_STATUS	4173
		33	0000'	CF	EB	0007D	BLBS	AED_L_STATUS, 8\$		
16		0000'	CF	03	E1	00082	BBC	#3, AED_L_FLAGS, 6\$	4176	
				01	DD	00088	PUSHL	#1		
				15	DD	0008A	PUSHL	#21		
	00000000G	00		02	FB	0008C	CALLS	#2, SCR\$ERASE_PAGE		
				01	DD	00093	PUSHL	#1		
				15	DD	00095	PUSHL	#21		
	00000000G	00		02	FB	00097	CALLS	#2, SCR\$SET_CURSOR		
			0000'	CF	DD	0009E	6\$:	PUSHL	AED_L_STATUS	

03	00000000G	00	01	FB	000A2	CALLS	#1, LIB\$SIGNAL		
	0000'	CF	03	EO	000A9	BBS	#3, AED_L_FLAGS, 78		
			02B2	31	000AF	BRW	398		
			029E	31	000B2	BRW	388		
			08	AE	D4	000B5	78:		
	E8	AD	00070200	8F	D0	000B8	88:		
	EC	AD	0000'	CF	D0	000C0			
			08	AE	9F	000C6	98:		
				7E	7C	000C9			
			E8	AD	9F	000CB			
			0000'	CF	9F	000CE			
			0000'	CF	9F	000D2			
			0000'	CF	3C	000D6			
	00000000G	00	07	FB	000DB	CALLS	#7, SYS\$CHANGE ACL		
	0000'	CF	50	D0	000E2	MOVL	R0, AED_L_STATUS		
		76	50	E8	000E7	BLBS	R0, 158		4197
	000009D0	8F	50	D1	000EA	CMPL	R0, #2512		4200
			07	13	000F1	BEQL	108		
	000009E0	8F	50	D1	000F3	CMPL	R0, #2528		4201
			03	12	000FA	108:			
			030C	31	000FC	BNEQ	118		
16	0000'	CF	03	E1	000FF	118:			
			01	DD	00105	BRW	478		
			15	DD	00107	PUSHL	#1		
	00000000G	00	02	FB	00109	PUSHL	#21		
			01	DD	00110	CALLS	#2, SCR\$ERASE_PAGE		
			15	DD	00112	PUSHL	#1		
	00000000G	00	02	FB	00114	PUSHL	#21		
			7E	D4	0011B	CALLS	#2, SCR\$SET_CURSOR		
			0000'	CF	DD	0011D	CLRL	-(SP)	
			0000'	CF	9F	00121	PUSHL	AED_L_STATUS	
			01	DD	00125	PUSHAB	AED_Q_OBJNAM		
			08	DD	00127	PUSHL	#1		
	00000000G	00	05	FB	0012D	PUSHL	#18157746		
11	0000'	CF	03	E1	00134	CALLS	#5, LIB\$SIGNAL		
		7E	0000'	CF	9A	0013A	#3, AED_L_FLAGS, 138		
		7E	0000'	CF	9A	0013F	AED_B_COLUMN, -(SP)		
	00000000G	00	02	FB	00144	MOVZBL	AED_B_LINE, -(SP)		
02	0000'	CF	03	ED	0014B	CALLS	#2, SCR\$SET_CURSOR		
			09	18	00152	CMPL	R0, #3, AED_L_WORSTERR, #2		
	0000'	CF	08	D0	00154	148:			
			0221	31	0015D	BGEQ	148		
			0000'	CF	D0	00160	MOVZBL	#18157746, AED_L_WORSTERR	
50	0000'	CF	00000200	8F	C1	00165	BRW	418	4204
		50	59	D1	0016F	158:			
			03	1F	00172	168:			
			FF4F	31	00174	178:			
			69	95	00177	188:			
			F9	13	00179				
			0C	AE	9F	0017B	TSTB	(ACE_POINTER)	4213
	04	AE	69	9A	0017E	BEQL	178		
			04	AE	9F	00182	PUSHAB	ACE_NEWADDR	4214
	00000000G	00	02	FB	00185	MOVZBL	(ACE_POINTER), 4(SP)		
		58	50	D0	0018C	PUSHL	4(SP)		
		0A	58	E9	0018F	CALLS	#2, LIB\$GET_VM		
		50	69	9A	00192	MOVL	R0, VM_STATUS		
50	00	6E	00	2C	00195	BLBC	VM_STATUS, 198		
						MOVZBL	(ACE_POINTER), R0		
						MOVCS	#0, (TSP), #0, R0, @ACE_NEWADDR		

PC	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418	Op419
----	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

	0000'	CF	58	DO	0027E	29\$:	MOVL	VM STATUS, AED_L STATUS	
		33	CF	E8	00283		BLBS	AED_L STATUS, 32\$	4244
16	0000'	CF	03	E1	00288		BBC	#3, AED_L_FLAGS, 30\$	4247
			01	DD	0028E		PUSHL	#1	
	00000000G	00	15	DD	00290		PUSHL	#21	
			02	FB	00292		CALLS	#2, SCR\$ERASE_PAGE	
			01	DD	00299		PUSHL	#1	
	00000000G	00	15	DD	0029B		PUSHL	#21	
			02	FB	0029D		CALLS	#2, SCR\$SET CURSOR	
	00000000G	00	0000'	CF	DD 002A4	30\$:	PUSHL	AED_L STATUS	
03	00000000G	00	01	FB	002A8		CALLS	#1, LTB\$SIGNAL	
	0000'	CF	03	E0	002AF		BBS	#3, AED_L_FLAGS, 31\$	
			FF2C	31	002B5		BRW	22\$	
			FF18	31	002B8	31\$:	BRW	21\$	
	08	58	10	AE	DO 002BB	32\$:	MOVL	NEW TEXT_LINE, R8	4253
	OC	A8	57	B0	002BF		MOVW	LINE_SEG_SIZE, 8(R8)	
14	AB	A8	OC	AE	DO 002C3		MOVL	ACE_NEWADDR, 12(R8)	4254
	0000'	6A	56	28	002C8		MOVW3	ACE_TEXT_SIZE, (FIRST CHAR), 20(R8)	4255
		DF	68	0E	002CD		INSQUE	(R8), @AED_Q LINETABLE+4	4256
			0000'	CF	D5 002D2		TSTL	AED_L_FIRSTLINE	4257
				06	12 002D6		BNEQ	33\$	
	0000'	CF	10	AE	DO 002DB		MOVL	NEW TEXT_LINE, AED_L_FIRSTLINE	
	0000'	CF	10	AE	DO 002DE	33\$:	MOVL	NEW TEXT_LINE, AED_L_LASTLINE	4258
		5A	01	AB	9E 002E4		MOVAB	1(RT1), FIRST_CHAR	4259
52		56	57	C3	002E8		SUBL3	LINE_SEG_SIZE, ACE_TEXT_SIZE, R2	4260
		56	FF	A2	9E 002EC		MOVAB	-1(R2), ACE_TEXT_SIZE	
			FF55	31	002F0		BRW	26\$	4238
			56	D5	002F3	34\$:	TSTL	ACE_TEXT_SIZE	4262
			03	14	002F5		BGTR	35\$	
			00C6	31	002F7		BRW	44\$	
			10	AE	9F 002FA	35\$:	PUSHAB	NEW TEXT_LINE	4266
	04	52	14	A6	9E 002FD		MOVAB	20(R6), R2	
		AE	52	DO	00301		MOVL	R2, 4(SP)	
	00000000G	00	04	AE	9F 00305		PUSHAB	4(SP)	
			02	FB	00308		CALLS	#2, LIB\$GET VM	
		58	50	DO	0030F		MOVL	R0, VM STATUS	
		07	58	E9	00312		BLBC	VM STATUS, 36\$	
52	00	6E	00	2C	00315		MOVW5	#0, (SP), #0, R2, @NEW_TEXT_LINE	
			10	BE	0031A				
	0000'	CF	58	DO	0031C	36\$:	MOVL	VM STATUS, AED_L STATUS	
		71	0000'	CF	E8 00321		BLBS	AED_L STATUS, 42\$	4267
16	0000'	CF	03	E1	00326		BBC	#3, AED_L_FLAGS, 37\$	4270
			01	DD	0032C		PUSHL	#1	
			15	DD	0032E		PUSHL	#21	
	00000000G	00	02	FB	00330		CALLS	#2, SCR\$ERASE_PAGE	
			01	DD	00337		PUSHL	#1	
			15	DD	00339		PUSHL	#21	
	00000000G	00	02	FB	0033B		CALLS	#2, SCR\$SET CURSOR	
			0000'	CF	DD 00342	37\$:	PUSHL	AED_L STATUS	
	00000000G	00	01	FB	00346		CALLS	#1, LTB\$SIGNAL	
11	0000'	CF	03	E1	0034D		BBC	#3, AED_L_FLAGS, 39\$	
		7E	0000'	CF	9A 00353	38\$:	MOVZBL	AED_B_COLUMN, -(SP)	
		7E	0000'	CF	9A 00358		MOVZBL	AED_B_LINE, -(SP)	
	00000000G	00	02	FB	0035D		CALLS	#2, SCR\$SET CURSOR	
		50	0000'	CF	DO 00364	39\$:	MOVL	AED_L STATUS, R0	
		07	50	93	00369		BITB	R0, #7	
			13	13	0036C		BEQL	41\$	

51	51	0000'	50	CF	03	00	EF	0036E	EXTZV	#0, #3, R0, R1	:	
					03	00	ED	00373	CMPZV	#0, #3, AED_L_WORSTERR, R1	:	
		0000'		CF		05	18	0037A	BGEQ	41\$:	
		0000'		CF	2008	50	D0	0037C	40\$:	MOVL	R0, AED_L_WORSTERR	4272
					0000'	8F	AA	00381	41\$:	BICW2	#8200, AED_L_FLAGS+1	4273
	50	0000'		CF	10000000	94	00388		CLRB	TERM CHAR	4274	
						8F	C9	0038C	BISL3	#268435456, AED_L_WORSTERR, R0	4276	
						04	00396		RET		4277	
			58		10	AE	D0	00397	42\$:	MOVL	NEW_TEXT_LINE, R8	4278
		08	A8			56	B0	0039B		MOVW	ACE-TEXT-SIZE, B(R8)	4279
		OC	A8		OC	AE	D0	0039F		MOVL	ACE-NEWADDR, 12(R8)	4280
14	AB	0000'	6A			56	28	003A4		MOVC3	ACE-TEXT-SIZE, (FIRST CHAR), 20(R8)	4281
			DF			68	0E	003A9		INSQUE	(R8), @AED_Q_LINETABLE+4	4282
					0000'	CF	D5	003AE		TSTL	AED_L_FIRSTLINE	4283
						06	12	003B2		BNEQ	43\$	4284
		0000'		CF	10	AE	D0	003B4		MOVL	NEW_TEXT_LINE, AED_L_FIRSTLINE	4285
		0000'		CF	10	AE	D0	003BA	43\$:	MOVL	NEW_TEXT_LINE, AED_L_LASTLINE	4286
					0000'	CF	D0	003C0	44\$:	MOVL	AED_L_FIRSTLINE, RT	4287
		0A	A1			01	88	003C5		BISB2	#1, 10(R1)	4288
27		03	A9			02	E0	003C9		BBS	#2, 3(ACE POINTER), 45\$	4289
			50		01	A9	9A	003CE		MOVZBL	1(ACE POINTER), R0	4290
			01			50	91	003D2		CMPB	R0, #1	4291
						22	13	003D5		BEQL	46\$	4292
			02			50	91	003D7		CMPB	R0, #2	4293
						1D	13	003DA		BEQL	46\$	4294
			03			50	91	003DC		CMPB	R0, #3	4295
						18	13	003DF		BEQL	46\$	4296
			04			50	91	003E1		CMPB	R0, #4	4297
						13	13	003E4		BEQL	46\$	4298
			05			50	91	003E6		CMPB	R0, #5	4299
						0E	13	003E9		BEQL	46\$	4300
			06			50	91	003EB		CMPB	R0, #6	4301
						09	13	003EE		BEQL	46\$	4302
			09			50	91	003F0		CMPB	R0, #9	4303
						04	13	003F3		BEQL	46\$	4304
		0A	A1			10	88	003F5	45\$:	BISB2	#16, 10(R1)	4305
		50			0000'	CF	D0	003F9	46\$:	MOVL	AED_L_LASTLINE, R0	4306
		0A	A0			02	88	003FE		BISB2	#2, 10(R0)	4307
		50				69	9A	00402		MOVZBL	(ACE POINTER), R0	4308
		59				50	C0	00405		ADDL2	R0, ACE_POINTER	4309
						31	00408		BRW	16\$	4310	
					0000'	CF	9F	0040B	47\$:	PUSHAB	AED_A_ACLBUFFER	4311
		04	AE		0200	8F	3C	0040F		MOVZWL	#512, 4(SP)	4312
					04	AE	9F	00415		PUSHAB	4(SP)	4313
		00000000G	00			02	FB	00418		CALLS	#2, LIB\$FREE VM	4314
			50		0000'	CF	9E	0041F		MOVAB	AED_Q_LINETABLE, R0	4315
			50		0000'	CF	D1	00424		CMPL	AED_Q_LINETABLE, R0	4316
						3E	12	00429		BNEQ	48\$	4317
		0000'		CF	4020	8F	A8	0042B		BISW2	#16416, AED_L_FLAGS	4318
					0000'	CF	B4	00432		CLRW	SEGMENT SIZE	4319
					0000'	CF	B4	00436		CLRW	AED_W_TOTALSIZE	4320
		0000'		DF	0000'	CF	0E	0043A		INSQUE	AED-T_CURLINE, @AED_Q_LINETABLE+4	4321
			50		0000'	CF	9E	00441		MOVAB	AED-T_CURLINE, R0	4322
		0000'		CF		50	D0	00446		MOVL	R0, AED_L_LASTLINE	4323
		0000'		CF		50	D0	0044B		MOVL	R0, AED_L_FIRSTLINE	4324
		0A	A0			01	B0	00450		MOVW	#1, 10(R0)	4325
					0000'	CF	D4	00454		CLRL	AED_L_CURACE	4326

			0000'	CF	95	00458	TSTB	AED_L_FLAGS+1	4312		
				6C	18	0045C	BGEQ	52\$			
			0000'	CF	94	0045E	CLRB	AED_B_ACETYPE	4315		
		0000'	CF	08	8A	00462	BICB2	#8, AED_L_FLAGS+2	4316		
				68	11	00467	BRB	53\$	4317		
			0000'	CF	DD	00469	PUSHL	AED_Q_LINETABLE	4323		
		0000G	CF	01	FB	0046D	CALLS	#1, AED_COPSEGMENT			
		0000'	CF	0E	00472	INSQUE	AED_T_CURLINE, AED_Q_LINETABLE	4324			
			0000'	CF	9E	00479	MOVAB	AED_T_CURLINE, R0	4325		
		0000'	CF	50	DD	0047E	MOVL	R0, AED_L_LASTLINE			
		0000'	CF	50	DD	00483	MOVL	R0, AED_L_FIRSTLINE			
			0000'	CF	08	A1	80	00488	MOVL	AED_L_FIRSTLINE, R1	4326
				08	A1	80	0048D	MOVW	8(R1), AED_W_TOTALSIZE		
		0000'	CF	50	DD	00493	MOVL	AED_L_LASTLINE, R0	4327		
23		0A	A0	01	E0	00498	BBS	#1, 10(R0), 51\$			
			52	0000'	CF	9E	0049D	MOVAB	AED_T_CURLINE, R2	4330	
			52		50	D1	004A2	CMPL	R0, R2		
					05	12	004A5	BNEQ	50\$		
		0000'	CF	60	DD	004A7	MOVL	(R0), AED_L_LASTLINE	4331		
		0000'	CF	0000'	DF	DD	004AC	50\$: MOVL	AED_L_LASTLINE, AED_L_LASTLINE	4332	
			50	0000'	CF	DD	004B3	MOVL	AED_C_LASTLINE, R0	4333	
		0000'	CF	08	A0	A0	004B8	ADDW2	8(R0), AED_W_TOTALSIZE		
				D8	11	004BE	BRB	49\$	4327		
		0000'	CF	0C	A1	DD	004C0	51\$: MOVL	12(R1), AED_L_CURACE	4335	
			0000'	CF	95	004C6	TSTB	AED_L_FLAGS+1	4336		
				16	18	004CA	BGEQ	54\$			
		0000'	CF	08	88	004CC	BISB2	#8, AED_L_FLAGS+2	4339		
			0000'	CF	9F	004D1	PUSHAB	BUFFER INDEX	4340		
		0000G	CF	01	FB	004D5	CALLS	#1, AED_SELECTFIELD			
0000'	CF	0000'	CF	01	81	004DA	ADDB3	#1, BUFFER INDEX, AED_B_COLUMN	4341		
		0000'	CF	0000'	CF	DD	004E2	54\$: MOVL	AED_Q_LINETABLE, AED_C_BEGINLINE	4344	
		0000'	CF	01	90	004E9	MOVB	#1, AED_B_SAVE_LIN	4345		
		0000'	CF	0000'	CF	90	004EE	MOVB	AED_B_COLUMN, AED_B_SAVE_COL	4346	
				01	DD	004F5	55\$: PUSHL	#1	4351		
				01	DD	004F7	PUSHL	#1			
		00000000G	00	02	FB	004F9	CALLS	#2, SCR\$ERASE_PAGE			
				14	DD	00500	PUSHL	#20	4352		
				01	DD	00502	PUSHL	#1			
		00000000G	00	02	FB	00504	CALLS	#2, SCR\$SET_SCROLL			
		06	0000'	CF	E8	0050B	BLBS	AED_L_FLAGS, 56\$	4353		
09		0000'	CF	01	E1	00510	BBC	#1, AED_L_FLAGS, 57\$			
			0000'	CF	9F	00516	56\$: PUSHAB	P.AAG	4354		
		0000G	CF	01	FB	0051A	CALLS	#1, AED_PUTOUTPUT			
		0000'	CF	01	DD	0051F	57\$: MOVL	#1, TEMP_LINE	4355		
		10	AE	0000'	CF	DD	00524	MOVL	AED_L_BEGINLINE, NEW_TEXT_LINE	4356	
			53	0000'	CF	DD	0052A	MOVL	TEMP_LINE, R3	4359	
				01	DD	0052F	58\$: PUSHL	#1			
				53	DD	00531	PUSHL	R3			
		00000000G	00	02	FB	00533	CALLS	#2, SCR\$SET_CURSOR			
			10	AE	DD	0053A	MOVL	NEW_TEXT_LINE, R2	4360		
		0000'	CF	08	A2	80	0053E	MOVW	8(R2), ECHO_DESC		
		0000'	CF	14	A2	9E	00544	MOVAB	20(R2), ECHO_DESC+4	4361	
			0000'	CF	9F	0054A	PUSHAB	ECHO_DESC	4362		
		0000G	CF	01	FB	0054E	CALLS	#1, AED_PUTOUTPUT			
			0000'	CF	D6	00553	INCL	TEMP_LINE	4363		
04		0A	A2	03	E1	00557	BBC	#3, 10(R2), 59\$	4364		
		10	AE	62	DD	0055C	MOVL	(R2), NEW_TEXT_LINE	4365		

AED\$MAIN
V04-000

ACT_REFRESH - refresh the display

0 5
15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 B11ss-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 147
(31)

10	AE	10	BE	D0	00560	59\$:	MOVL	NEW_TEXT_LINE, NEW_TEXT_LINE	:	4366
	53	0000'	CF	D0	00565		MOVL	TEMP_LINE, R3	:	4368
	14		53	D1	0056A		CMPL	R3, #20	:	
			0B	14	0056D		BGTR	60\$:	
	50	0000'	CF	9E	0056F		MOVAB	AED_Q LINETABLE, R0	:	4369
	50	10	AE	D1	00574		CMPL	NEW_TEXT_LINE, R0	:	
			B5	12	00578		BNEQ	58\$:	
	7E	0000'	CF	9A	0057A	60\$:	MOVZBL	AED_B_SAVE_COL, -(SP)	:	4370
	7E	0000'	CF	9A	0057F		MOVZBL	AED_B_SAVE_LIN, -(SP)	:	
00000000G	00		02	FB	00584		CALLS	#2, SCR\$SET_CURSOR	:	
0000'	CF	2008'	8F	AA	0058B		BICW2	#8200, AED_E_FLAGS+1	:	4372
		0000'	CF	94	00592		CLRB	TERM_CHAR	:	4373
	50		01	D0	00596		MOVL	#1, R0	:	4374
				04	00599		RET		:	4376

: Routine Size: 1434 bytes. Routine Base: \$CODE\$ + 28CF

ACT_ENTER - enter the current ACE

```
3956 4377 1 %SBTTL 'ACT_ENTER - enter the current ACE'
3957 4378 1 ROUTINE ACT_ENTER =
3958 4379 1
3959 4380 1 ++
3960 4381 1
3961 4382 1 FUNCTIONAL DESCRIPTION:
3962 4383 1
3963 4384 1 This routine updates the object's ACL with the newly modified ACE.
3964 4385 1 The cursor is left positioned at the first character of the next
3965 4386 1 ACE.
3966 4387 1
3967 4388 1 CALLING SEQUENCE:
3968 4389 1 ACT_ENTER ()
3969 4390 1
3970 4391 1 INPUT PARAMETERS:
3971 4392 1 none
3972 4393 1
3973 4394 1 IMPLICIT INPUTS:
3974 4395 1 OWN storage
3975 4396 1
3976 4397 1 OUTPUT PARAMETERS:
3977 4398 1 none
3978 4399 1
3979 4400 1 IMPLICIT OUTPUTS:
3980 4401 1 none
3981 4402 1
3982 4403 1 ROUTINE VALUE:
3983 4404 1 1 if successful
3984 4405 1 error status otherwise
3985 4406 1
3986 4407 1 SIDE EFFECTS:
3987 4408 1 The line segment table is updated as necessary, ACE line pointers
3988 4409 1 are updated, and the object's ACL is updated as necessary.
3989 4410 1
3990 4411 1 --
3991 4412 1
3992 4413 2 BEGIN
3993 4414 2
3994 4415 2 BUFFER_INDEX = 0;
3995 4416 2 AED_B_COLUMN = 1;
3996 4417 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
3997 4418 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
3998 4419 2 IF AED_L_FLAGS[AED_V_MODIFIED]
3999 4420 2 OR AED_L_FLAGS[AED_V_INSERT]
4000 4421 2 OR AED_L_FLAGS[AED_V_INSERTTEXT]
4001 4422 2 THEN
4002 4423 2 BEGIN
4003 4424 2 FINISH_ACE ();
4004 4425 2 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
4005 4426 2 AED_COMPRESS ();
4006 4427 2 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
4007 4428 2 IF NOT AED_L_STATUS
4008 4429 2 THEN
4009 4430 2 BEGIN
4010 4431 2 AED_L_FLAGS[AED_V_ACERROR] = 1;
4011 4432 2 AED_POSITION (.AED_L_FIRSTLINE);
4012 4433 2 AED_COPSEGMENT (.AED_L_FIRSTLINE);
```

```
4013 4434 4      INSQUE (AED T CURLINE[LINE L FLINK],
4014 4435 4      .AED L FIRSTLINE[LINE L BLINK]);
4015 4436 4      IF .AED L LASTLINE EQL .AED L FIRSTLINE
4016 4437 4      THEN AED L LASTLINE = AED T CURLINE;
4017 4438 4      IF .AED L BEGINLINE EQL .AED L FIRSTLINE
4018 4439 4      THEN AED L BEGINLINE = AED T CURLINE;
4019 4440 4      AED L FIRSTLINE = AED T CURLINE;
4020 4441 4      IF .AED L FIRSTLINE NEQ .AED L LASTLINE
4021 4442 4      AND .AED L FLAGS[AED V ENDACL]
4022 4443 4      THEN AED L FLAGS[AED V ENDACL] = 0;
4023 4444 4      BUFFER INDEX = 0;
4024 4445 4      AED B COLUMN = 1;
4025 4446 4      AED SET CURSOR (.AED B LINE, .AED B COLUMN);
4026 4447 4      AED L FLAGS[AED V GO[KEY]] = 0;
4027 4448 4      AED L FLAGS[AED V ACTIONKEY] = 0;
4028 4449 4      TERM CHAR = 0;
4029 4450 4      RETURN 1;
4030 4451 4      END;
4031 4452 4      AED L FLAGS[AED V MODIFIED] = AED L FLAGS[AED V INSERT] = 0;
4032 4453 4      END;
4033 4454 4
4034 4455 4      ! Set up the display for the next line.
4035 4456 4
4036 4457 4      AED POSITION (.AED L LASTLINE[LINE L FLINK]);
4037 4458 4      IF .AED L LASTLINE[LINE L FLINK] EQL AED Q LINETABLE[LINE L FLINK]
4038 4459 4      THEN
4039 4460 4      BEGIN
4040 4461 4      AED L FLAGS[AED V ENDACL] = 1;
4041 4462 4      AED L FLAGS[AED V INSERTTEXT] = 1;
4042 4463 4      AED W TOTALSIZE = SEGMENT SIZE = 0;
4043 4464 4      INSQUE (AED T CURLINE[LINE L FLINK],
4044 4465 4      .AED Q LINETABLE[LINE L BLINK]);
4045 4466 4      AED L FIRSTLINE = AED L LASTLINE = AED T CURLINE;
4046 4467 4      AED L FIRSTLINE[LINE Q FLAGS] = LINE M BEGINACE;
4047 4468 4      AED L CURACE = 0;
4048 4469 4      IF .AED L FLAGS[AED V PROMPT]
4049 4470 4      THEN
4050 4471 4      BEGIN
4051 4472 4      AED B ACETYPE = 0;
4052 4473 4      AED L FLAGS[AED V NOITEMSEL] = 0;
4053 4474 4      AED SELECTFIELD (BUFFER INDEX);
4054 4475 4      ECHO DESC[DSCSW LENGTH] = .AED T CURLINE[LINE W SIZE];
4055 4476 4      ECHO DESC[DSCSA POINTER] = AED T CURLINE[LINE T TEXT];
4056 4477 4      SCRSET CURSOR (.AED B LINE, 1);
4057 4478 4      AED PUTOUTPUT (ECHO DESC);
4058 4479 4      SCRERASE LINE (.AED B LINE, .SEGMENT SIZE + 1);
4059 4480 4      AED B COLUMN = .BUFFER INDEX + 1;
4060 4481 4      AED SET CURSOR (.AED B LINE, .AED B COLUMN);
4061 4482 4      END;
4062 4483 4      END
4063 4484 4      ELSE
4064 4485 4      BEGIN
4065 4486 4      AED COPSEGMENT (.AED L LASTLINE[LINE L FLINK]);
4066 4487 4      INSQUE (AED T CURLINE[LINE L FLINK], .AED L LASTLINE[LINE L FLINK]);
4067 4488 4      AED L FIRSTLINE = AED L LASTLINE = AED T CURLINE;
4068 4489 4      AED W TOTALSIZE = .AED L FIRSTLINE[LINE Q SIZE];
4069 4490 4      UNTIL .AED L LASTLINE[LINE V ENDACE]
```

```
4070 4491 3 DO
4071 4492 4 BEGIN
4072 4493 4 IF .AED_L_LASTLINE EQLA AED_T_CURLINE
4073 4494 4 THEN AED_C_LASTLINE = .AED_C_LASTLINE[LINE_L_FLINK];
4074 4495 4 AED_L_LASTLINE = .AED_L_LASTLINE[LINE_L_FLINK];
4075 4496 4 AED_W_TOTALSIZE = .AED_W_TOTALSIZE + .AED_L_LASTLINE[LINE_W_SIZE];
4076 4497 4 END;
4077 4498 4 AED_L_CURACE = .AED_L_FIRSTLINE[LINE_L_BINACE];
4078 4499 4 IF .AED_L_FLAGS[AED_V_PROMPT]
4079 4500 4 THEN
4080 4501 4 BEGIN
4081 4502 4 AED_L_FLAGS[AED_V_NOITEMSEL] = 1;
4082 4503 4 AED_SELECTFIELD(BUFFER_INDEX);
4083 4504 4 AED_B_COLUMN = .BUFFER_INDEX + 1;
4084 4505 4 END;
4085 4506 4 END;
4086 4507 4 AED_L_FLAGS[AED_V_FIRSTCHAR] = 1;
4087 4508 4 IF .AED_B_COLUMN GTR .SEGMENT_SIZE + 1
4088 4509 4 THEN BUFFER_INDEX = .SEGMENT_SIZE
4089 4510 4 ELSE BUFFER_INDEX = .AED_B_COLUMN - 1;
4090 4511 4 AED_SET_CURSOR(.AED_B_LINE, .BUFFER_INDEX + 1);
4091 4512 4 AED_L_FLAGS[AED_V_GOLDREY] = 0;
4092 4513 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
4093 4514 4 TERM_CHAR = 0;
4094 4515 4 RETURN 1;
4095 4516 4
4096 4517 1 END;
```

! End of routine ACT_ENTER

001C 00000 ACT_ENTER:

54	0000'	CF	9E	00002	WORD	Save R2,R3,R4	4378
53	0000'	CF	9E	00007	MOVAB	BUFFER_INDEX, R4	
		64	D4	0000C	MOVAB	AED_L_FLAGS, R3	
20	A3	01	90	0000E	CLRL	BUFFER_INDEX	4415
01	A3	08	8A	00012	MOVB	#1, AED_B_COLUMN	4416
0000G	CF	00	FB	00016	BICB2	#8, AED_L_FLAGS+1	4417
18	A4	50	D0	0001B	CALLS	#0, AED_REPSEGMENT	4418
		63	95	0001F	MOVL	R0, NEW_TEXT_LINE	
		0D	19	00021	TSTB	AED_L_FLAGS	4419
08	01	05	E0	00023	BLSS	1\$	
03	01	06	E0	00028	BBS	#5, AED_L_FLAGS+1, 1\$	4420
		00B1	31	0002D	BBS	#6, AED_L_FLAGS+1, 1\$	4421
		00	FB	00030	BRW	6\$	
0000V	CF	40	8F	8A	CALLS	#0, FINISH_ACE	4424
01	A3				BICB2	#64, AED_L_FLAGS+1	4425
0000G	CF	02C4	C3	3C	CALLS	#0, AED_COMPRESS	4426
	7E		01	FB	MOVZWL	AED_W_TOTALSIZE, -(SP)	4427
0000G	CF		50	D0	CALLS	#1, AED_UPDATEACL	
008C	C3	008C	C3	E8	MOVL	R0, AED_L_STATUS	
	59	40	8F	88	BLBS	AED_L_STATUS, 5\$	4428
	63	40	A3	DD	BISB2	#64, AED_L_FLAGS	4431
			01	FB	PUSHL	AED_L_FIRSTLINE	4432
0000G	CF	40	A3	DD	CALLS	#1, AED_POSITION	
					PUSHL	AED_L_FIRSTLINE	4433

0000G	CF		01	FB	00062	CALLS	#1, AED COPSEGMENT	4435
	50	40	A3	DO	00067	MOVL	AED_L_FIRSTLINE, RO	4436
04	B0	00B0	C3	OE	00068	INSQUE	AED_T_CURLINE, @4(RO)	4437
40	A3	44	A3	D1	00071	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	4438
			06	12	00076	BNEQ	28	4439
44	A3	00B0	C3	9E	00078	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	4440
40	A3	48	A3	D1	0007E	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	4441
			06	12	00083	BNEQ	38	4442
48	A3	00B0	C3	9E	00085	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	4443
40	A3	00B0	C3	9E	00088	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	4444
44	A3	40	A3	D1	00091	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	4445
			07	13	00096	BEQL	48	4446
03	63		05	E1	00098	BBC	#5, AED_L_FLAGS, 48	4447
	63		20	8A	0009C	BICB2	#32, AED_C_FLAGS	4448
			64	D4	0009F	CLRL	BUFFER INDEX	4449
20	A3		01	90	000A1	MOVAB	#1, AED_B_COLUMN	4450
	7E	20	A3	9A	000A5	MOVZBL	AED_B_COLUMN, -(SP)	4451
			0127	31	000A9	BRW	168	4452
	63	2080	8F	AA	000AC	BICW2	#8320, AED_L_FLAGS	4453
		44	B3	DD	000B1	PUSHL	@AED_L_LASTLINE	4454
0000G	CF		01	FB	000B4	CALLS	#1, AED POSITION	4455
	50	30	A3	9E	000B9	MOVAB	AED_Q_LINETABLE, RO	4456
	50	44	B3	D1	000BD	CMPL	@AED_C_LASTLINE, RO	4457
			03	13	000C1	BEQL	78	4458
			00B2	31	000C3	BRW	98	4459
	63	4020	8F	A8	000C6	BISW2	#16416, AED_L_FLAGS	4460
		00B8	C3	B4	000CB	CLRW	SEGMENT SIZE	4461
		02C4	C3	B4	000CF	CLRW	AED_W_TOTALSIZE	4462
34	B3	00B0	C3	OE	000D3	INSQUE	AED_T_CURLINE, @AED_Q_LINETABLE+4	4463
	50	00B0	C3	9E	000D9	MOVAB	AED_T_CURLINE, RO	4464
44	A3		50	DO	000DE	MOVL	RO, AED_L_LASTLINE	4465
40	A3		50	DO	000E2	MOVL	RO, AED_L_FIRSTLINE	4466
0A	A0		01	B0	000E6	MOVW	#1, 10(RO)	4467
		3C	A3	D4	000EA	CLRL	AED_L_CURACE	4468
		01	A3	95	000ED	TSTB	AED_L_FLAGS+1	4469
			54	18	000F0	BGEQ	88	4470
		00A8	C3	94	000F2	CLRB	AED_B_ACETYPE	4471
02	A3		08	8A	000F6	BICB2	#8, AED_L_FLAGS+2	4472
			54	DD	000FA	PUSHL	R4	4473
0000G	CF		01	FB	000FC	CALLS	#1, AED SELECTFIELD	4474
04	A4	00B8	C3	B0	00101	MOVW	AED_T_CURLINE+8, ECHO_DESC	4475
08	A4	00C4	C3	9E	00107	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	4476
			01	DD	0010D	PUSHL	#1	4477
	7E	24	A3	9A	0010F	MOVZBL	AED_B_LINE, -(SP)	4478
00000000G	00		02	FB	00113	CALLS	#2, -STR\$SET_CURSOR	4479
		04	A4	9F	0011A	PUSHAB	ECHO_DESC	4480
0000G	CF		01	FB	0011D	CALLS	#1, AED_PUTOUTPUT	4481
	7E	00B8	C3	3C	00122	MOVZWL	SEGMENT_SIZE, -(SP)	4482
			6E	D6	00127	INCL	(SP)	4483
	7E	24	A3	9A	00129	MOVZBL	AED_B_LINE, -(SP)	4484
00000000G	00		02	FB	0012D	CALLS	#2, -STR\$ERASE LINE	4485
20	A3		01	81	00134	ADDB3	#1, BUFFER INDEX, AED_B_COLUMN	4486
	64	20	A3	9A	00139	MOVZBL	AED_B_COLUMN, -(SP)	4487
	7E	24	A3	9A	0013D	MOVZBL	AED_B_LINE, -(SP)	4488
0000G	CF		02	FB	00141	CALLS	#2, AED_SET_CURSOR	4489
			67	11	00146	BRB	138	4490
		44	B3	DD	00148	PUSHL	@AED_L_LASTLINE	4491

0000G	CF	01	FB	0014B	CALLS	#1, AED_COPSEGMENT	...	4487
44	B3	00B0	C3	0E	00150	INSQUE	AED_T_CURLINE, @AED_L_LASTLINE	4488
	50	00B0	C3	9E	00156	MOVAB	AED_T_CURLINE, R0	
44	A3		50	D0	0015B	MOVL	R0, AED_L_LASTLINE	
40	A3		50	D0	0015F	MOVL	R0, AED_L_FIRSTLINE	
	52	40	A3	D0	00163	MOVL	AED_L_FIRSTLINE, R2	4489
02C4	C3	08	A2	B0	00167	MOVW	8(R2), AED_W_TOTALSIZE	
	51	44	A3	D0	0016D	MOVL	AED_L_LASTLINE, R1	4490
1F	0A		01	E0	00171	BBS	#1, -10(R1) 12\$	
	50	00B0	C3	9E	00176	MOVAB	AED_T_CURLINE, R0	4493
	50		51	D1	0017B	CMPL	R1, R0	
			04	12	0017E	BNEQ	11\$	
44	A3		61	D0	00180	MOVL	(R1), AED_L_LASTLINE	4494
44	A3	44	B3	D0	00184	MOVL	@AED_L_LASTLINE, AED_L_LASTLINE	4495
	51	44	A3	D0	00189	MOVL	AED_L_LASTLINE, R1	4496
02C4	C3	08	A1	A0	0018D	ADDW2	8(R1), AED_W_TOTALSIZE	
			DC	11	00193	BRB	10\$	4490
3C	A3	0C	A2	D0	00195	MOVL	12(R2), AED_L_CURACE	4498
		01	A3	95	0019A	TSTB	AED_L_FLAGS+1	4499
			10	18	0019D	BGEQ	13\$	
02	A3		08	88	0019F	BISB2	#8, AED_L_FLAGS+2	4502
			54	D0	001A3	PUSHL	R4	4503
0000G	CF	01	FB	001A5	CALLS	#1, AED_SELECTFIELD		
20	A3		01	81	001AA	ADDB3	#1, BUFFER_INDEX, AED_B_COLUMN	4504
	01		10	88	001AF	BISB2	#16, AED_L_FLAGS+1	4507
	50	00B8	C3	3C	001B3	MOVZWL	SEGMENT_SIZE, R0	4508
			50	D6	001B8	INCL	R0	
50	20	A3	00	ED	001BA	CMPZV	#0, #8, AED_B_COLUMN, R0	
			07	15	001C0	BLEQ	14\$	
	64	00B8	C3	3C	001C2	MOVZWL	SEGMENT_SIZE, BUFFER_INDEX	4509
			06	11	001C7	BRB	15\$	
	64	20	A3	9A	001C9	MOVZBL	AED_B_COLUMN, BUFFER_INDEX	4510
			64	D7	001CD	DECL	BUFFER_INDEX	
7E	64		01	C1	001CF	ADDL3	#1, BUFFER_INDEX, -(SP)	4511
	7E	24	A3	9A	001D3	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	001D7	CALLS	#2, AED_SET_CURSOR	
01	A3	2008	8F	AA	001DC	BICW2	#8200, AED_L_FLAGS+1	4513
		28	A4	94	001E2	CLRB	TERM_CHAR	4514
	50		01	D0	001E5	MOVL	#1, R0	4515
			04	001E8	RET			4517

Routine Size: 489 bytes. Routine Base: \$CODE\$ + 2E69

ACT_INSERT - insert an ACE

```
4098 4518 1 %SBTTL 'ACT_INSERT - insert an ACE'
4099 4519 1 ROUTINE ACT_INSERT =
4100 4520 1
4101 4521 1 ++
4102 4522 1
4103 4523 1 FUNCTIONAL DESCRIPTION:
4104 4524 1
4105 4525 1     This routine is called when it is desired to insert a new ACE at
4106 4526 1     a random position in the ACL.
4107 4527 1
4108 4528 1 CALLING SEQUENCE:
4109 4529 1     ACT_INSERT ()
4110 4530 1
4111 4531 1 INPUT PARAMETERS:
4112 4532 1     none
4113 4533 1
4114 4534 1 IMPLICIT INPUTS:
4115 4535 1     OWN storage
4116 4536 1
4117 4537 1 OUTPUT PARAMETERS:
4118 4538 1     none
4119 4539 1
4120 4540 1 IMPLICIT OUTPUTS:
4121 4541 1     none
4122 4542 1
4123 4543 1 ROUTINE VALUE:
4124 4544 1     1 if successful
4125 4545 1     error status otherwise
4126 4546 1
4127 4547 1 SIDE EFFECTS:
4128 4548 1     The line segment table is updated as necessary, ACE line pointers
4129 4549 1     are updated, and the object's ACL is updated as necessary.
4130 4550 1
4131 4551 1 --
4132 4552 1
4133 4553 2 BEGIN
4134 4554 2
4135 4555 2 IF NOT .AED_L_FLAGS[AED_V_ENDACL]
4136 4556 2 AND NOT .AED_C_FLAGS[AED_V_INSERTTEXT]
4137 4557 2 AND NOT .AED_L_FLAGS[AED_V_INSERT]
4138 4558 2 THEN
4139 4559 2     BEGIN
4140 4560 2     NEW_TEXT_LINE = AED_REPSEGMENT ();
4141 4561 2     IF .AED_C_FLAGS[AED_V_MODIFIED]
4142 4562 2     THEN
4143 4563 4         BEGIN
4144 4564 4         FINISH ACE ();
4145 4565 4         IF .AED_L_FLAGS[AED_V_PROMPT]
4146 4566 4         AND .AED_C_FLAGS[AED_V_FIRSTCHAR]
4147 4567 4         THEN
4148 4568 5             BEGIN
4149 4569 5             NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
4150 4570 5             AED_W_TOTALSIZE = 0;
4151 4571 4             END;
4152 4572 4             AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
4153 4573 4             IF .AED_W_TOTALSIZE EQL 0
4154 4574 4             THEN NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_L_BLINK];
```

```
4155 4575 4 AED_COMPRESS ();
4156 4576 4 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
4157 4577 4 IF NOT .AED_L_STATUS
4158 4578 4 THEN
4159 4579 4 BEGIN
4160 4580 4 AED_L_FLAGS[AED_V_ACERROR] = 1;
4161 4581 4 AED_POSITION (.AED_L_FIRSTLINE);
4162 4582 4 AED_COPSEGMENT (.AED_L_FIRSTLINE);
4163 4583 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK],
4164 4584 4 .AED_C_FIRSTLINE[LINE_L_BLINK]);
4165 4585 4 IF .AED_L_LASTLINE EQL .AED_L_FIRSTLINE
4166 4586 4 THEN AED_C_LASTLINE = AED_T_CURLINE;
4167 4587 4 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
4168 4588 4 THEN AED_C_BEGINLINE = AED_T_CURLINE;
4169 4589 4 AED_L_FIRSTLINE = AED_T_CURLINE;
4170 4590 4 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
4171 4591 4 AND .AED_C_FLAGS[AED_V_ENDACL]
4172 4592 4 THEN AED_L_FLAGS[AED_V_ENDACL] = 0;
4173 4593 4 BUFFER_INDEX = 0;
4174 4594 4 AED_B_COLUMN = 1;
4175 4595 4 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
4176 4596 4 AED_L_FLAGS[AED_V_GODREY] = 0;
4177 4597 4 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
4178 4598 4 TERM_CHAR = 0;
4179 4599 4 RETURN 1;
4180 4600 4 END;
4181 4601 4 AED_L_FLAGS[AED_V_MODIFIED] = AED_L_FLAGS[AED_V_INSERT] = 0;
4182 4602 4 END;
4183 4603 4 AED_COMPRESS ();
4184 4604 4 AED_POSITION (.AED_L_FIRSTLINE);
4185 4605 4 AED_L_CURACE = .AED_C_FIRSTLINE[LINE_L_BINACE];
4186 4606 4 AED_W_TOTALSIZE = SEGMENT_SIZE = 0;
4187 4607 4 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_L_FIRSTLINE[LINE_L_BLINK]);
4188 4608 4 IF .AED_L_BEGINLINE EQL .AED_L_FIRSTLINE
4189 4609 4 THEN AED_C_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
4190 4610 4 AED_L_FIRSTLINE = AED_L_LASTLINE = AED_T_CURLINE;
4191 4611 4 AED_L_FIRSTLINE[LINE_O_FLAGS] = LINE_M_BEGINACE;
4192 4612 4 AED_POSITION (AED_T_CURLINE);
4193 4613 4
4194 4614 4 ! Now repaint the display. This is done by either scrolling down and repainting
4195 4615 4 the first part of the display or repainting from the current position to the
4196 4616 4 end of the display (or the end of the ACL). This is necessary to echo the
4197 4617 4 text from the split portion of the line.
4198 4618 4
4199 4619 4 IF .AED_B_LINE LEQ 10
4200 4620 4 THEN
4201 4621 4 BEGIN
4202 4622 4 SCR$SET_CURSOR (1,1); ! **** TEMP ****
4203 4623 4 SCR$DOWN_SCROLL ();
4204 4624 4 NEW_TEXT_LINE = .AED_L_BEGINLINE;
4205 4625 4 INCR J FROM 1 TO .AED_B_LINE
4206 4626 4 DO
4207 4627 4 BEGIN
4208 4628 4 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
4209 4629 4 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
4210 4630 4 SCR$SET_CURSOR (J, 1);
4211 4631 4 AED_PUTOUTPUT (ECHO_DESC);
```



```
4212 4632 3 SCR$ERASE LINE (.J, .ECHO_DESC[DSC$W_LENGTH] + 1);
4213 4633 3 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_C_FLINK];
4214 4634 3 END;
4215 4635 3 ELSE
4216 4636 3 BEGIN
4217 4637 3 NEW_TEXT_LINE = AED_T_CURLINE;
4218 4638 3 INCR J FROM .AED_B_LINE TO 20
4219 4639 3 DO
4220 4640 3 BEGIN
4221 4641 3 ECHO_DESC[DSC$W_LENGTH] = .NEW_TEXT_LINE[LINE_W_SIZE];
4222 4642 3 ECHO_DESC[DSC$A_POINTER] = NEW_TEXT_LINE[LINE_T_TEXT];
4223 4643 3 SCR$SET CURSOR T.J, 1;
4224 4644 3 AED_PUTOUTPUT (ECHO_DESC);
4225 4645 3 SCR$ERASE LINE (.J, .ECHO_DESC[DSC$W_LENGTH] + 1);
4226 4646 3 NEW_TEXT_LINE = .NEW_TEXT_LINE[LINE_C_FLINK];
4227 4647 3 IF .NEW_TEXT_LINE EQ[A AED_Q_LINETABLE[LINE_L_FLINK] THEN EXITLOOP;
4228 4648 3 END;
4229 4649 3 END;
4230 4650 3 BUFFER_INDEX = 0;
4231 4651 3 AED_B_COLUMN = 1;
4232 4652 3 IF .AED_L_FLAGS[AED_V_PROMPT]
4233 4653 3 THEN
4234 4654 3 BEGIN
4235 4655 3 AED_B_ACETYPE = 0;
4236 4656 3 AED_L_FLAGS[AED_V_NOITEMSEL] = 0;
4237 4657 3 AED_SELECTFIELD (BUFFER_INDEX);
4238 4658 3 ECHO_DESC[DSC$W_LENGTH] = .AED_T_CURLINE[LINE_W_SIZE];
4239 4659 3 ECHO_DESC[DSC$A_POINTER] = AED_T_CURLINE[LINE_T_TEXT];
4240 4660 3 SCR$SET CURSOR T.AED_B_LINE, 1;
4241 4661 3 AED_PUTOUTPUT (ECHO_DESC);
4242 4662 3 SCR$ERASE LINE (.AED_B_LINE, .SEGMENT_SIZE + 1);
4243 4663 3 AED_B_COLUMN = .BUFFER_INDEX + 1;
4244 4664 3 END;
4245 4665 3 AED_SET CURSOR (.AED_B_LINE, .AED_B_COLUMN);
4246 4666 3 AED_L_FLAGS[AED_V_FIRSTCHAR] = 1;
4247 4667 3 AED_L_FLAGS[AED_V_INSERTTEXT] = 1;
4248 4668 3 AED_L_FLAGS[AED_V_INSERT] = 1;
4249 4669 3 END;
4250 4670 3 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
4251 4671 3 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
4252 4672 3 TERM_CHAR = 0;
4253 4673 3 RETURN 1;
4254 4674 3
4255 4675 3
4256 4676 1 END;
```

! End of routine ACT_INSERT

03FC 0000 ACT_INSERT:

59	0000G	CF	9E	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9
58	0000G	CF	9E	00007	MOVAB	AED_PUTOUTPUT, R9
57	00000000G	00	9E	0000C	MOVAB	AED_POSITION, R8
56	00000000G	00	9E	00013	MOVAB	SCR\$ERASE LINE, R7
55	0000'	CF	9E	0001A	MOVAB	SCR\$SET CURSOR, R6
						NEW_TEXT_LINE, R5

4519

03		54	0000'	CF	9E	0001F	MOVAB	AED_L_FLAGS, R4		
		64		05	E1	00024	BBC	#5, AED_L_FLAGS, 28	4555	
			01FB	31	00028	18:	BRW	198		
F8	01	A4		06	E0	0002B	BBS	#6, AED_L_FLAGS+1, 18	4556	
F3	01	A4		05	E0	00030	BBS	#5, AED_L_FLAGS+1, 18	4557	
	0000G	CF		00	FB	00035	CALLS	#0, AED-REPSEGMENT	4560	
		65		50	D0	0003A	MOVL	R0, NEW-TEXT_LINE		
				64	95	0003D	TSTB	AED_L_FLAGS	4561	
				03	19	0003F	BLSS	38		
			00AB	31	00041		BRW	108		
	0000V	CF		00	FB	00044	CALLS	#0, FINISH_ACE	4564	
			01	A4	95	00049	TSTB	AED_L_FLAGS+1	4565	
				10	18	0004C	BGEQ	48		
0B	01	A4		04	E1	0004E	BBC	#4, AED_L_FLAGS+1, 48	4566	
		50		65	D0	00053	MOVL	NEW-TEXT_LINE, R0	4569	
	0A	A0		04	88	00056	BISB2	#4, -10(R0)		
			02C4	C4	84	0005A	CLRW	AED_W_TOTALSIZE	4570	
			40	8F	8A	0005E	BICB2	#64, AED_L_FLAGS+1	4572	
	01	A4	02C4	C4	B5	00063	TSTW	AED_W_TOTALSIZE	4573	
				07	12	00067	BNEQ	58		
		50		65	D0	00069	MOVL	NEW-TEXT_LINE, R0	4574	
		65	04	A0	D0	0006C	MOVL	4(R0), NEW-TEXT_LINE		
	0000G	CF		00	FB	00070	CALLS	#0, AED_COMPRESS	4575	
		7E	02C4	C4	3C	00075	MOVZWL	AED_W_TOTALSIZE, -(SP)	4576	
	0000G	CF		01	FB	0007A	CALLS	#1, AED_UPDATEACL		
	008C	C4		50	D0	0007F	MOVL	R0, AED_L_STATUS		
		61	008C	C4	E8	00084	BLBS	AED_L_STATUS, 98	4577	
		64	40	8F	88	00089	BISB2	#64, AED_L_FLAGS	4580	
			40	A4	DD	0008D	PUSHL	AED_L_FIRSTLINE	4581	
		68		01	FB	00090	CALLS	#1, AED_POSITION		
			40	A4	DD	00093	PUSHL	AED_L_FIRSTLINE	4582	
	0000G	CF		01	FB	00096	CALLS	#1, AED_COPSEGMENT		
		50	40	A4	D0	0009B	MOVL	AED_L_FIRSTLINE, R0	4584	
	04	B0	00B0	C4	0E	0009F	INSQUE	AED_T_CURLINE, 84(R0)		
	40	A4	44	A4	D1	000A5	CMPL	AED_L_LASTLINE, AED_L_FIRSTLINE	4585	
				06	12	000AA	BNEQ	68		
	44	A4	00B0	C4	9E	000AC	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	4586	
	40	A4	48	A4	D1	000B2	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	4587	
				06	12	000B7	BNEQ	78		
	48	A4	00B0	C4	9E	000B9	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	4588	
	40	A4	00B0	C4	9E	000BF	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	4589	
	44	A4	40	A4	D1	000C5	CMPL	AED_L_FIRSTLINE, AED_L_LASTLINE	4590	
				07	13	000CA	BEQL	88		
03		64		05	E1	000CC	BBC	#5, AED_L_FLAGS, 88	4591	
		64		20	8A	000D0	BICB2	#32, AED_L_FLAGS	4592	
			E8	A5	D4	000D3	CLRL	BUFFER_INDEX	4593	
	20	A4		01	90	000D6	MOVB	#1, AED_B_COLUMN	4594	
		7E	20	A4	9A	000DA	MOVZBL	AED_B_COLUMN, -(SP)	4595	
		7E	24	A4	9A	000DE	MOVZBL	AED_B_LINE, -(SP)		
	0000G	CF		02	FB	000E2	CALLS	#2, AED_SET_CURSOR		
			013C	31	000E7		BRW	198	4596	
		64	2080	8F	AA	000EA	BICW2	#8320, AED_L_FLAGS	4601	
	0000G	CF		00	FB	000EF	CALLS	#0, AED_COMPRESS	4603	
			40	A4	DD	000F4	PUSHL	AED_L_FIRSTLINE	4604	
		68		01	FB	000F7	CALLS	#1, AED_POSITION		
		50	40	A4	D0	000FA	MOVL	AED_L_FIRSTLINE, R0	4605	
	3C	A4	0C	A0	D0	000FE	MOVL	12(R0), AED_L_CURACE		

		00B8	C4	B4	00103	CLRW	SEGMENT SIZE	4606
		02C4	C4	B4	00107	CLRW	AED_W_TOTALSIZE	4607
04	B0	00B0	C4	0E	0010B	INSQUE	AED_T_CURLINE, B4(R0)	4608
40	A4	48	A4	D1	00111	CMPL	AED_L_BEGINLINE, AED_L_FIRSTLINE	4609
			06	12	00116	BNEG	11\$	4610
48	A4	00B0	C4	9E	00118	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	4611
	50	00B0	C4	9E	0011E	MOVAB	AED_T_CURLINE, R0	4612
44	A4		50	DD	00123	MOVL	R0, AED_L_LASTLINE	4619
40	A4		50	DD	00127	MOVL	R0, AED_L_FIRSTLINE	4622
0A	A0		01	B0	0012B	MOVW	#1, 10(R0)	4623
		00B0	C4	9F	0012F	PUSHAB	AED_T_CURLINE	4624
	68		01	FB	00133	CALLS	#1, AED_POSITION	4625
	0A	24	A4	91	00136	CMPB	AED_B_LINE, #10	4628
			48	1A	0013A	BGTRU	14\$	4629
			01	DD	0013C	PUSHL	#1	4630
			01	DD	0013E	PUSHL	#1	4631
	66		02	FB	00140	CALLS	#2, SCR\$SET_CURSOR	4632
00000000G	00		00	FB	00143	CALLS	#0, SCR\$DOWN_SCROLL	4633
	65	48	A4	DD	0014A	MOVL	AED_L_BEGINLINE, NEW_TEXT_LINE	4634
	53	24	A4	9A	0014E	MOVZBL	AED_B_LINE, R3	4635
			52	D4	00152	CLRL	J	4636
			28	11	00154	BRB	13\$	4637
	50		65	DD	00156	MOVL	NEW TEXT LINE, R0	4638
EC	A5	08	A0	B0	00159	MOVW	8(R0), ECHO_DESC	4639
FO	A5	14	A0	9E	0015E	MOVAB	20(R0), ECHO_DESC+4	4640
			01	DD	00163	PUSHL	#1	4641
	66		52	DD	00165	PUSHL	J	4642
			02	FB	00167	CALLS	#2, SCR\$SET_CURSOR	4643
	69	EC	A5	9F	0016A	PUSHAB	ECHO_DESC	4644
	7E	EC	01	FB	0016D	CALLS	#1, AED_PUTOUTPUT	4645
			A5	3C	00170	MOVZWL	ECHO_DESC, -(SP)	4646
			6E	D6	00174	INCL	(SP)	4647
			52	DD	00176	PUSHL	J	4648
	67		02	FB	00178	CALLS	#2, SCR\$ERASE_LINE	4649
	75		95	DD	0017B	MOVL	NEW TEXT LINE, NEW_TEXT_LINE	4650
D4	52		53	F3	0017E	AOBLEQ	R3, J, 12\$	4651
			45	11	00182	BRB	17\$	4652
	65	00B0	C4	9E	00184	MOVAB	AED_T_CURLINE, NEW_TEXT_LINE	4653
	52		65	DD	00189	MOVL	NEW_TEXT_LINE, R2	4654
	53	24	A4	9A	0018C	MOVZBL	AED_B_LINE, J	4655
			53	D7	00190	DECL	J	4656
			31	11	00192	BRB	16\$	4657
EC	A5	08	A2	B0	00194	MOVW	8(R2), ECHO_DESC	4658
FO	A5	14	A2	9E	00199	MOVAB	20(R2), ECHO_DESC+4	4659
			01	DD	0019E	PUSHL	#1	4660
	66		53	DD	001A0	PUSHL	J	4661
			02	FB	001A2	CALLS	#2, SCR\$SET_CURSOR	4662
	69	EC	A5	9F	001A5	PUSHAB	ECHO_DESC	4663
	7E	EC	01	FB	001AB	CALLS	#1, AED_PUTOUTPUT	4664
			A5	3C	001AB	MOVZWL	ECHO_DESC, -(SP)	4665
			6E	D6	001AF	INCL	(SP)	4666
			53	DD	001B1	PUSHL	J	4667
	67		02	FB	001B3	CALLS	#2, SCR\$ERASE_LINE	4668
	75		95	DD	001B6	MOVL	NEW TEXT LINE, NEW_TEXT_LINE	4669
	52		65	DD	001B9	MOVL	NEW TEXT LINE, R2	4670
	50	30	A4	9E	001BC	MOVAB	AED_Q_LINETABLE, R0	4671
	50		52	D1	001C0	CMPL	R2, R0	4672

CB	53		04	13	001C3	BEQL	17\$		
		E8	14	F3	001C5	AOBLEQ	#20, J, 15\$	4639	
20	A4		A5	D4	001C9	CLRL	BUFFER INDEX	4651	
		01	01	90	001CC	MOVB	#1, AED_B_COLUMN	4652	
			A4	95	001D0	TSTB	AED_L_FLAGS+1	4653	
		00A8	3F	18	001D3	BGEQ	18\$		
			C4	94	001D5	CLRB	AED_B_ACETYPE	4656	
02	A4		08	8A	001D9	BICB2	#8, AED_L_FLAGS+2	4657	
		E8	A5	9F	001DD	PUSHAB	BUFFER INDEX	4658	
0000G	CF		01	FB	001E0	CALLS	#1, AED_SELECTFIELD		
EC	A5	00B8	C4	80	001E5	MOVW	AED_T_CURLINE+8, ECHO_DESC	4659	
FO	A5	00C4	C4	9E	001EB	MOVAB	AED_T_CURLINE+20, ECHO_DESC+4	4660	
			01	DD	001F1	PUSHL	#1	4661	
	7E	24	A4	9A	001F3	MOVZBL	AED_B_LINE, -(SP)		
66			02	FB	001F7	CALLS	#2, SCRSET_CURSOR		
		EC	A5	9F	001FA	PUSHAB	ECHO_DESC	4662	
	69		01	FB	001FD	CALLS	#1, AED_PUTOUTPUT		
	7E	00B8	C4	3C	00200	MOVZWL	SEGMENT_SIZE, -(SP)	4663	
			6E	D6	00205	INCL	(SP)		
	7E	24	A4	9A	00207	MOVZBL	AED_B_LINE, -(SP)		
20	A4	E8	02	FB	0020B	CALLS	#2, SCRERASE_LINE		
			01	81	0020E	ADDB3	#1, BUFFER INDEX, AED_B_COLUMN	4664	
	7E	20	A4	9A	00214	MOVZBL	AED_B_COLUMN, -(SP)	4666	
	7E	24	A4	9A	00218	MOVZBL	AED_B_LINE, -(SP)		
0000G	CF		02	FB	0021C	CALLS	#2, AED_SET_CURSOR		
01	A4	70	8F	88	00221	BISB2	#112, AED_L_FLAGS+1	4669	
01	A4	2008	8F	AA	00226	BICW2	#8200, AED_L_FLAGS+1	4672	
		10	A5	94	0022C	CLRB	TERM_CHAR	4673	
	50		01	D0	0022F	MOVL	#1, R0	4674	
			04	00	00232	RET		4676	

; Routine Size: 563 bytes, Routine Base: \$CODE\$ + 3052

ACT_EXIT - Leave the ACL editor

```
4258 4677 1 XSBTTL 'ACT_EXIT - Leave the ACL editor'
4259 4678 1 ROUTINE ACT_EXIT (QUIT) =
4260 4679 1
4261 4680 1 ++
4262 4681 1
4263 4682 1 FUNCTIONAL DESCRIPTION:
4264 4683 1
4265 4684 1 This routine handles two ACL editor actions. If the action is a
4266 4685 1 QUIT, then any existing journal file is closed (but not deleted),
4267 4686 1 any unentered ACE is left untouched, and the session is terminated.
4268 4687 1
4269 4688 1 If the action is an EXIT, the object's ACL is updated to reflect the
4270 4689 1 current state. This done by first deleting any existing ACL, and
4271 4690 1 adding the current in core one.
4272 4691 1
4273 4692 1 CALLING SEQUENCE:
4274 4693 1 ACT_EXIT (ARG1)
4275 4694 1
4276 4695 1 INPUT PARAMETERS:
4277 4696 1 ARG1: 1 = terminate the session with the object's ACL untouched
4278 4697 1 0 = terminate the session and update the object's ACL
4279 4698 1
4280 4699 1 IMPLICIT INPUTS:
4281 4700 1 OWN storage
4282 4701 1
4283 4702 1 OUTPUT PARAMETERS:
4284 4703 1 none
4285 4704 1
4286 4705 1 IMPLICIT OUTPUTS:
4287 4706 1 none
4288 4707 1
4289 4708 1 ROUTINE VALUE:
4290 4709 1 1 if successful
4291 4710 1 error status otherwise
4292 4711 1
4293 4712 1 SIDE EFFECTS:
4294 4713 1 The line segment table is updated as necessary, ACE line pointers
4295 4714 1 are updated, and the object's ACL is updated as necessary.
4296 4715 1
4297 4716 1 --
4298 4717 1
4299 4718 2 BEGIN
4300 4719 2
4301 4720 2 LOCAL
4302 4721 2 ATR_ARGLIST : BLOCKVECTOR [2, ITMSS_ITEM, BYTE], ! ACL attributes
4303 4722 2 ACL_CONTEXT, ! ACL context for $CHANGE_ACL
4304 4723 2 DUMMY_ACE : $BLOCK [ACLSS_DELALENT], ! Dummy ACE for delete
4305 4724 2 CURRENT_LINE : REF $BLOCK; ! Current line segment address
4306 4725 2
4307 4726 2 ! Determine if this is a QUIT or EXIT.
4308 4727 2
4309 4728 2 IF .QUIT
4310 4729 2 THEN
4311 4730 2 BEGIN
4312 4731 2 SIGNAL (AEDS_NOCHANGE);
4313 4732 2 RETURN 0;
4314 4733 2 END;
```

```
4315 4734 2
4316 4735 2 NEW_TEXT_LINE = AED_REPSEGMENT ();
4317 4736 2 IF .AED_C_FLAGS[AED_V_MODIFIED]
4318 4737 2 OR .AED_L_FLAGS[AED_V_INSERT]
4319 4738 2 OR .AED_L_FLAGS[AED_V_INSERTTEXT]
4320 4739 2 THEN
4321 4740 2 BEGIN
4322 4741 2 FINISH_ACE ();
4323 4742 2 IF .AED_L_FLAGS[AED_V_PROMPT]
4324 4743 2 AND .AED_C_FLAGS[AED_V_FIRSTCHAR]
4325 4744 2 THEN
4326 4745 2 BEGIN
4327 4746 2 NEW_TEXT_LINE[LINE_V_DUMMY] = 1;
4328 4747 2 AED_W_TOTALSIZE = 0;
4329 4748 2 END;
4330 4749 2 AED_COMPRESS ();
4331 4750 2 AED_L_STATUS = AED_UPDATEACL (.AED_W_TOTALSIZE);
4332 4751 2 IF NOT .AED_L_STATUS
4333 4752 2 THEN
4334 4753 2 BEGIN
4335 4754 2 AED_L_FLAGS[AED_V_ACERROR] = 1;
4336 4755 2 AED_POSITION (.AED_L_FIRSTLINE);
4337 4756 2 AED_COSEGMENT (.AED_L_FIRSTLINE);
4338 4757 2 INSQUE (AED_T_CURLINE[LINE_L_FLINK], .AED_L_FIRSTLINE[LINE_L_BLINK]);
4339 4758 2 IF .AED_L_LASTLINE EQL .AED_C_FIRSTLINE
4340 4759 2 THEN AED_C_LASTLINE = AED_T_CURLINE;
4341 4760 2 IF .AED_C_BEGINLINE EQL .AED_L_FIRSTLINE
4342 4761 2 THEN AED_C_BEGINLINE = AED_T_CURLINE;
4343 4762 2 AED_L_FIRSTLINE = AED_T_CURLINE;
4344 4763 2 IF .AED_L_FIRSTLINE NEQ .AED_L_LASTLINE
4345 4764 2 AND .AED_C_FLAGS[AED_V_ENDACL]
4346 4765 2 THEN
4347 4766 2 BEGIN
4348 4767 2 AED_L_FLAGS[AED_V_ENDACL] = 0;
4349 4768 2 AED_L_FLAGS[AED_V_INSERTTEXT] = 0;
4350 4769 2 END;
4351 4770 2 BUFFER_INDEX = 0;
4352 4771 2 AED_B_COLUMN = 1;
4353 4772 2 AED_SET_CURSOR (.AED_B_LINE, .AED_B_COLUMN);
4354 4773 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
4355 4774 2 AED_L_FLAGS[AED_V_ACTIONKEY] = 0;
4356 4775 2 TERM_CHAR = 0;
4357 4776 2 RETURN 1;
4358 4777 2 END;
4359 4778 2 END;
4360 4779 2 AED_L_FLAGS[AED_V_GOLDKEY] = 0;
4361 4780 2
4362 4781 2 ! Now for the fun part. Because the real ACL on the object hasn't been
4363 4782 2 ! touched, it is necessary to update it at this time. This is done by
4364 4783 2 ! first deleting the object's ACL, and then applying the ACL as modified
4365 4784 2 ! by the user.
4366 4785 2
4367 4786 2 CHSFILL (0, 2*ITMSS_ITEM, ATR_ARGLIST);
4368 4787 2 ACL_CONTEXT = 0;
4369 4788 2
4370 4789 2 ! Now delete the entire ACL. This will catch all but the protected ACEs.
4371 4790 2
```

```
4372 4791 2 ATR_ARGLIST[0, ITMSW_ITMCD] = ACL$C_DELETEACL;
4373 4792 ATR_ARGLIST[0, ITMSW_BUFSIZ] = 12;
4374 4793 ATR_ARGLIST[0, ITMSL_BUFADR] = DUMMY_ACE;
4375 4794 AED_L_STATUS = $CHANGE_ACL (CHAN = .AED_W_OBJCHAN,
4376 4795 OBJTYP = AED_C_OBJTYP,
4377 4796 OBJNAM = AED_Q_OBJNAM,
4378 4797 ITMLST = ATR_ARGLIST,
4379 4798 CONTXT = ACL_CONTEXT);
4380 4799 IF NOT .AED_L_STATUS
4381 4800 THEN
4382 4801 BEGIN
4383 4802 AED_B_OPTIONS[AED_V_KEEPPJNL] = 1; ! Keep the journal file
4384 4803 RETURN 0;
4385 4804 END;
4386 4805
4387 4806 ! Now delete any protected ACEs remaining in the ACL.
4388 4807
4389 4808 WHILE 1
4390 4809 DO
4391 4810 BEGIN
4392 4811 ATR_ARGLIST[0, ITMSW_ITMCD] = ACL$C_READACE;
4393 4812 ATR_ARGLIST[0, ITMSW_BUFSIZ] = ACL$S_READACE;
4394 4813 ATR_ARGLIST[0, ITMSL_BUFADR] = DUMMY_ACE;
4395 4814 ACL_CONTEXT = 0;
4396 4815 AED_L_STATUS = $CHANGE_ACL (CHAN = .AED_W_OBJCHAN,
4397 4816 OBJTYP = AED_C_OBJTYP,
4398 4817 OBJNAM = AED_Q_OBJNAM,
4399 4818 ITMLST = ATR_ARGLIST,
4400 4819 CONTXT = ACL_CONTEXT);
4401 4820 IF NOT .AED_L_STATUS
4402 4821 THEN
4403 4822 BEGIN
4404 4823 IF .AED_L_STATUS EQL $$$_ACLEMPY
4405 4824 OR .AED_L_STATUS EQL $$$_NOMOREACE
4406 4825 THEN EXIT[00];
4407 4826 AED_B_OPTIONS[AED_V_KEEPPJNL] = 1; ! Keep the journal file
4408 4827 SIGNAL (.AED_L_STATOS);
4409 4828 RETURN 0;
4410 4829 END;
4411 4830 ATR_ARGLIST[0, ITMSW_ITMCD] = ACL$C_DELAENT;
4412 4831 ATR_ARGLIST[0, ITMSW_BUFSIZ] = .DUMMY_ACE[ACESB_SIZE];
4413 4832 ATR_ARGLIST[0, ITMSL_BUFADR] = DUMMY_ACE;
4414 4833 AED_L_STATUS = $CHANGE_ACL (CHAN = .AED_W_OBJCHAN,
4415 4834 OBJTYP = AED_C_OBJTYP,
4416 4835 OBJNAM = AED_Q_OBJNAM,
4417 4836 ITMLST = ATR_ARGLIST,
4418 4837 CONTXT = ACL_CONTEXT);
4419 4838 IF NOT .AED_L_STATUS
4420 4839 THEN
4421 4840 BEGIN
4422 4841 AED_B_OPTIONS[AED_V_KEEPPJNL] = 1; ! Keep the journal file
4423 4842 SIGNAL (.AED_L_STATOS);
4424 4843 RETURN 0;
4425 4844 END;
4426 4845 END;
4427 4846
4428 4847 ! Now that the object's original ACL has been removed, update the ACL with the
```

```
4429 4848 2 ! one modified by the user.
4430 4849 2
4431 4850 2 CURRENT_LINE = .AED_Q LINETABLE[LINE_L FLINK];
4432 4851 2 UNTIL .CURRENT_LINE EQL AED_Q LINETABLE[LINE_L FLINK]
4433 4852 2 DO
4434 4853 2 BEGIN
4435 4854 2 IF .CURRENT_LINE[LINE_V BEGINACE]
4436 4855 2 AND .CURRENT_LINE[LINE_L BINACE] NEQ 0
4437 4856 2 THEN
4438 4857 2 BEGIN
4439 4858 2 ATR_ARGLIST[0, ITMSW_ITMCD] = ACL$C ADDACLENT;
4440 4859 2 ATR_ARGLIST[0, ITMSW_BUFSIZ] = $BBLOCK [.CURRENT_LINE[LINE_L BINACE], ACESB_SIZE];
4441 4860 2 ATR_ARGLIST[0, ITMSL_BUFADR] = .CURRENT_LINE[LINE_L BINACE];
4442 4861 2 ACL_CONTEXT = %X'00FFFFFF';
4443 4862 2 AED_L_STATUS = $CHANGE_ACL (CHAN = .AED_W OBJCHAN,
4444 4863 2 OBJTYP = AED_L OBJTYP,
4445 4864 2 OBJNAM = AED_Q OBJNAM,
4446 4865 2 ITMLST = ATR_ARGLIST,
4447 4866 2 CONTXT = ACL_CONTEXT);
4448 4867 2
4449 4868 2 IF NOT .AED_L_STATUS
4450 4869 2 THEN
4451 4870 2 BEGIN
4452 4871 2 AED_B OPTIONS[AED_V KEEPJNL] = 1; ! Keep the journal file
4453 4872 2 SIGNAL (.AED_L_STATUS);
4454 4873 2 RETURN 0;
4455 4874 2 END;
4456 4875 2 CURRENT_LINE = .CURRENT_LINE[LINE_L FLINK];
4457 4876 2 END;
4458 4877 2
4459 4878 2 SIGNAL (AED$_ACLUPDATED);
4460 4879 2
4461 4880 2 RETURN 0;
4462 4881 2
4463 4882 2 1 END; ! End of routine ACT_EXIT
```

OFFC 00000 ACT_EXIT:

	SB	00000000G	BF	DO	00002	WORD	Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11	4678
	5A	00000000G	00	9E	00009	MOVL	#AED\$ NOCHANGE, R11	
	59	00000000G	00	9E	00010	MOVAB	SYSSCHANGE_ACL, R10	
	58	00000000G	00	9E	00017	MOVAB	LIB\$SIGNAL, R9	
	57	00000000G	00	9E	0001E	MOVAB	SCR\$ERASE_PAGE, R8	
	56	00000000G	00	9E	0001E	MOVAB	SCR\$SET_CURSOR, R7	
	5E	0000	CF	9E	00025	MOVAB	AED_L_FLAGS, R6	
	41	FEE4	CE	9E	0002A	MOVAB	-284(SP), SP	
	66	04	AC	E9	0002F	BLBC	QUIT, 48	4728
OE			03	E1	00033	BBC	#3, AED_L_FLAGS, 18	4731
			01	DD	00037	PUSHL	#1	
			15	DD	00039	PUSHL	#21	
	68		02	FB	0003B	CALLS	#2, SCR\$ERASE_PAGE	
			01	DD	0003E	PUSHL	#1	
			15	DD	00040	PUSHL	#21	
	67		02	FB	00042	CALLS	#2, SCR\$SET_CURSOR	

00000000*	8F	14	A6	03	00	ED	00061	38:	CALLS	#1, LIBSSIGNAL	4732
					04	18	0006B	48:	BBC	#3, AED_L_FLAGS, 28	4735
					02	FB	00056		MOVZBL	AED_B_COLUMN, -(SP)	
					02	FB	00056		MOVZBL	AED_B_LINE, -(SP)	
					02	FB	00056		CALLS	#2, SCRSET CURSOR	
					02	FB	00056		TSTL	#<AED\$_NOCHANGE&7>	
					10	13	0005F	28:	BEQL	38	
					00	ED	00061		CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_NOCHANGE&7>	
					04	18	0006B		BGEQ	38	
					02	FB	00074	38:	MOVZBL	R11, AED_L_WORSTERR	4736
					00	FB	00074	48:	BRW	278	
					50	DO	00079		CALLS	#0, AED_REPSEGMENT	4737
					66	95	0007E		MOVZBL	RO, NEW_TEXT_LINE	4738
					0D	19	00080		TSTB	AED_L_FLAGS	
					05	E0	00082		BLSS	58	
					06	E0	00087		BBS	#5, AED_L_FLAGS+1, 58	4739
					00	A6	31 0008C		BBS	#6, AED_L_FLAGS+1, 58	4740
					00	FB	0008F	58:	BRW	108	
					12	18	00097		CALLS	#0, FINISH_ACE	4741
					04	E1	00099		TSTB	AED_L_FLAGS+1	4742
					04	88	000A3		BGEQ	68	
					04	B4	000A7		BBC	#4, AED_L_FLAGS+1, 68	4743
					00	FB	000AB	68:	MOVZBL	NEW_TEXT_LINE, RO	4744
					01	FB	000B5		BISB2	#4, -10(RO)	
					50	DO	000BA		CLRW	AED_W_TOTALSIZE	4745
					06	E8	000BF		CALLS	#0, AED_COMPRESS	4746
					08	8F	000C4		MOVZBL	AED_W_TOTALSIZE, -(SP)	4747
					01	FB	000CB		CALLS	#1, AED_UPDATEACL	4748
					01	FB	000CB		MOVZBL	AED_L_FIRSTLINE	4749
					01	FB	000D3		CALLS	#1, AED_POSITION	4750
					06	12	000E7		PUSHL	AED_L_FIRSTLINE	4751
					06	12	000F4		CALLS	#1, AED_COPSEGMENT	4752
					06	12	000F4		MOVZBL	AED_L_FIRSTLINE, RO	4753
					06	12	000F4		INSQUE	AED_T_CURLINE, 24(RO)	4754
					06	12	000F4		CMPZV	AED_L_LASTLINE, AED_L_FIRSTLINE	4755
					06	12	000F4		BNEQ	78	
					06	12	000F4		MOVAB	AED_T_CURLINE, AED_L_LASTLINE	4756
					06	12	000F4		CMPZV	AED_L_BEGINLINE, AED_L_FIRSTLINE	4757
					06	12	000F4		BNEQ	88	
					06	12	000F4		MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	4758
					06	12	000F4		MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	4759
					06	12	000F4		MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	4760
					06	12	000F4		CMPZV	AED_L_FIRSTLINE, AED_L_LASTLINE	4761
					06	12	000F4		BEQL	98	
					06	12	000F4		BBC	#5, AED_L_FLAGS, 98	4762
					06	12	000F4		BICW2	#16416, AED_L_FLAGS	4763
					06	12	000F4		CLRL	BUFFER_INDEX	4764
					06	12	000F4		MOVZBL	#1, AED_B_COLUMN	4765
					06	12	000F4		MOVZBL	AED_B_COLUMN, -(SP)	4766
					06	12	000F4		MOVZBL	AED_B_LINE, -(SP)	4767
					06	12	000F4		CALLS	#2, AED_SET CURSOR	4768
					06	12	000F4		BICW2	#8200, AED_C_FLAGS+1	4769
					06	12	000F4		CLRB	TERM_CHAR	4770
					06	12	000F4		MOVZBL	#1, RO	4771

18	00	01	A6	08	04	00134	RET			
			6E	00	8A	00135	BICB2	#8, AED_L_FLAGS+1	4779	
				AD	2C	00139	MOVC3	#0, (SPT, #0, #24, ATR_ARGLIST	4786	
				6E	D4	00140	CLRL	ACL_CONTEXT	4787	
		E8	AD	8F	DD	00142	MOVL	#393228, ATR_ARGLIST	4792	
		EC	AD	AE	9E	0014A	MOVAB	DUMMY_ACE, ATR_ARGLIST+4	4793	
				5E	DD	0014F	PUSHL	SP	4798	
				7E	7C	00151	CLRG	-(SP)		
				AD	9F	00153	PUSHAB	ATR_ARGLIST		
				OC	A6	00156	PUSHAB	AED_Q_OBJNAM		
				08	A6	00159	PUSHAB	AED_L_OBJTYP		
				78	A6	0015C	MOVZWL	AED_W_OBJCHAN, -(SP)		
		7E		07	FB	00160	CALLS	#7, SYSSCHANGE_ACL		
	008C	6A		50	DD	00163	MOVL	R0, AED_L_STATUS		
		C6		C6	E8	00168	BLBS	AED_L_STATUS, 11\$	4799	
		07		08	88	0016D	BISB2	#8, AED_B_OPTIONS	4802	
	04	A6		01	31	00171	BRW	27\$	4803	
				8F	DD	00174	MOVL	#590079, ATR_ARGLIST	4812	
		E8	AD	AE	9E	0017C	MOVAB	DUMMY_ACE, ATR_ARGLIST+4	4813	
		EC	AD	6E	D4	00181	CLRL	ACL_CONTEXT	4814	
				5E	DD	00183	PUSHL	SP	4819	
				7E	7C	00185	CLRG	-(SP)		
				AD	9F	00187	PUSHAB	ATR_ARGLIST		
				OC	A6	0018A	PUSHAB	AED_Q_OBJNAM		
				08	A6	0018D	PUSHAB	AED_L_OBJTYP		
				78	A6	00190	MOVZWL	AED_W_OBJCHAN, -(SP)		
		7E		07	FB	00194	CALLS	#7, SYSSCHANGE_ACL		
	008C	6A		50	DD	00197	MOVL	R0, AED_L_STATUS		
		C6		50	E8	0019C	BLBS	R0, 13\$	4820	
	000009D0	8F		50	D1	0019F	CMPL	R0, #2512	4823	
				66	13	001A6	BEQL	15\$		
	000009E0	8F		50	D1	001A8	CMPL	R0, #2528	4824	
				5D	13	001AF	BEQL	15\$		
		04	A6	08	88	001B1	BISB2	#8, AED_B_OPTIONS	4826	
			66	03	E0	001B5	BBS	#3, AED_L_FLAGS, 12\$	4827	
03				00B7	31	001B9	BRW	20\$		
				00A6	31	001BC	BRW	19\$		
		EA	AD	02	80	001BF	MOVW	#2, ATR_ARGLIST+2	4830	
		E8	AD	04	AE	9B	MOVZBW	DUMMY_ACE, ATR_ARGLIST	4831	
		EC	AD	04	AE	9E	MOVAB	DUMMY_ACE, ATR_ARGLIST+4	4832	
					5E	DD	PUSHL	SP	4837	
					7E	7C	CLRG	-(SP)		
					AD	9F	PUSHAB	ATR_ARGLIST		
					OC	A6	PUSHAB	AED_Q_OBJNAM		
					08	A6	PUSHAB	AED_L_OBJTYP		
					78	A6	MOVZWL	AED_W_OBJCHAN, -(SP)		
		7E		07	FB	001DA	CALLS	#7, SYSSCHANGE_ACL		
	008C	6A		50	DD	001E1	MOVL	R0, AED_L_STATUS		
		C6		C6	E8	001E6	BLBS	AED_L_STATUS, 11\$	4838	
		89		08	88	001EB	BISB2	#8, AED_B_OPTIONS	4841	
	04	A6		03	E1	001EF	BBC	#3, AED_L_FLAGS, 14\$	4842	
0E				01	DD	001F3	PUSHL	#1		
				15	DD	001F5	PUSHL	#21		
				02	FB	001F7	CALLS	#2, SCR\$ERASE_PAGE		
		68		01	DD	001FA	PUSHL	#1		
				15	DD	001FC	PUSHL	#21		

		67		02	FB	001FE		CALLS	#2, SCR\$SET_CURSOR		
			008C	C6	DD	00201	148:	PUSHL	AED_L_STATUS		
		69		01	FB	00205		CALLS	#1, LTB\$SIGNAL		
72		66		03	EO	00208		BBS	#3, AED_L_FLAGS, 218		
				7B	11	0020C		BRB	228		
		52	30	A6	DD	0020E	158:	MOVL	AED_Q_LINETABLE, CURRENT_LINE	4850	
		50	30	A6	9E	00212	168:	MOVAB	AED_Q_LINETABLE, RO	4851	
		50		52	D1	00216		CMPL	CURRENT_LINE, RO		
				03	12	00219		BNEQ	178		
				008E	31	0021B		BRW	248		
		03	0A	A2	EB	0021E	178:	BLBS	10(CURRENT_LINE), 188	4854	
				0081	31	00222		BRW	238		
			0C	A2	DS	00225	188:	TSTL	12(CURRENT_LINE)	4855	
				7C	13	00228		BEQL	238		
EA	AD			01	BO	0022A		MOVW	#1, ATR_ARGLIST+2	4858	
E8	AD	0C		B2	9B	0022E		MOVZBW	12(CURRENT_LINE), ATR_ARGLIST	4859	
EC	AD	0C		A2	DD	00233		MOVL	12(CURRENT_LINE), ATR_ARGLIST+4	4860	
	6E	00FFFFFF		8F	DD	00238		MOVL	#16777215, -ACL_CONTEXT	4861	
				5E	DD	0023F		PUSHL	SP	4866	
				7E	7C	00241		CLRG	-(SP)		
		E8	AD	9F	00243			PUSHAB	ATR_ARGLIST		
		0C	A6	9F	00246			PUSHAB	AED_Q_OBJNAM		
		08	A6	9F	00249			PUSHAB	AED_L_OBJTYP		
		78	A6	3C	0024C			MOVZWL	AED_W_OBJCHAN, -(SP)		
	7E			07	FB	00250		CALLS	#7, SYSS\$CHANGE_ACL		
008C	C6			50	DD	00253		MOVL	RO, AED_L_STATUS		
	49	008C		C6	EB	00258		BLBS	AED_L_STATUS, 238	4867	
	A6	04		08	88	0025D		BISB2	#8, AED_B_OPTIONS	4870	
OE	66			03	E1	00261		BBC	#3, AED_L_FLAGS, 208	4871	
				01	DD	00265	198:	PUSHL	#1		
				15	DD	00267		PUSHL	#21		
	68			02	FB	00269		CALLS	#2, SCR\$ERASE_PAGE		
				01	DD	0026C		PUSHL	#1		
				15	DD	0026E		PUSHL	#21		
	67			02	FB	00270		CALLS	#2, SCR\$SET_CURSOR		
		008C		C6	DD	00273	208:	PUSHL	AED_L_STATUS		
	69			01	FB	00277		CALLS	#1, LTB\$SIGNAL		
OB	66			03	E1	0027A		BBC	#3, AED_L_FLAGS, 228		
	7E	20		A6	9A	0027E	218:	MOVZBL	AED_B_COLUMN, -(SP)		
	7E	24		A6	9A	00282		MOVZBL	AED_B_LINE, -(SP)		
	67			02	FB	00286		CALLS	#2, SCR\$SET_CURSOR		
	50	008C		C6	DD	00289	228:	MOVL	AED_L_STATUS, RO		
	07			50	93	0028E		BITB	RO, #7		
				5F	13	00291		BEQL	278		
51		50		00	EF	00293		EXTZV	#0, #3, RO, R1		
51	14	A6		00	ED	00298		CMPZV	#0, #3, AED_L_WORSTERR, R1		
				52	18	0029E		BGEQ	278		
	14	A6		50	DD	002A0		MOVL	RO, AED_L_WORSTERR	4872	
				4C	11	002A4		BRB	278		
	52			62	DD	002A6	238:	MOVL	(CURRENT_LINE), CURRENT_LINE	4875	
			FF66	31	002A9			BRW	168	4851	
OE	66			03	E1	002AC	248:	BBC	#3, AED_L_FLAGS, 258	4878	
				01	DD	002B0		PUSHL	#1		
				15	DD	002B2		PUSHL	#21		
	68			02	FB	002B4		CALLS	#2, SCR\$ERASE_PAGE		
				01	DD	002B7		PUSHL	#1		
				15	DD	002B9		PUSHL	#21		

AED\$MAIN
V04-000

ACT_EXIT - Leave the ACL editor

15-Sep-1984 23:47:14
14-Sep-1984 11:52:29

VAX-11 B11sg-32 V4.0-742
[ACLEDT.SRC]AEDMAIN.B32;1

Page 166
(34)

			67	00000000G	02	FB	002B8	258:	CALLS	#2, SCR\$SET CURSOR
			69		8F	DD	002BE		PUSHL	#AED\$_ACLUPDATED
	0B		66		01	FB	002C4		CALLS	#1, LTB\$SIGNAL
			7E	20	03	E1	002C7		BBC	#3, AED_L_FLAGS, 268
			7E	24	A6	9A	002CB		MOVZBL	AED_B_COLUMN, -(SP)
			67		A6	9A	002CF		MOVZBL	AED_B_LINE, -(SP)
				00000000*	02	FB	002D3	268:	CALLS	#2, SCR\$SET CURSOR
					8F	D5	002D6		TSTL	#<AED\$_ACLUPDATED&7>
00000000*	8F				14	13	002DC		BEQL	278
	14	A6	03		00	ED	002DE		CMPZV	#0, #3, AED_L_WORSTERR, #<AED\$_ACLUPDATED&-7>
					08	18	002E8		BGEQ	278
			14	A6 00000000G	8F	D0	002EA	278:	MOVL	#AED\$_ACLUPDATED, AED_L_WORSTERR
					50	D4	002F2		CLRL	R0
					04	002F4			RET	

4882

; Routine Size: 757 bytes, Routine Base: \$CODE\$ + 3285

FINISH_ACE - Tie off the current ACE

```
4465 4883 1 %SBTTL 'FINISH_ACE - Tie off the current ACE'
4466 4884 1 ROUTINE FINISH_ACE : NOVALUE =
4467 4885 1
4468 4886 1 **
4469 4887 1
4470 4888 1 FUNCTIONAL DESCRIPTION:
4471 4889 1
4472 4890 1     This routine ties off the current ACE. I.e., it adds a final right
4473 4891 1     paren if necessary.
4474 4892 1
4475 4893 1 CALLING SEQUENCE:
4476 4894 1     FINISH_ACE ()
4477 4895 1
4478 4896 1 INPUT PARAMETERS:
4479 4897 1     none
4480 4898 1
4481 4899 1 IMPLICIT INPUTS:
4482 4900 1     OWN storage
4483 4901 1
4484 4902 1 OUTPUT PARAMETERS:
4485 4903 1     none
4486 4904 1
4487 4905 1 IMPLICIT OUTPUTS:
4488 4906 1     none
4489 4907 1
4490 4908 1 ROUTINE VALUE:
4491 4909 1     none
4492 4910 1
4493 4911 1 SIDE EFFECTS:
4494 4912 1     none
4495 4913 1
4496 4914 1 --
4497 4915 1
4498 4916 2 BEGIN
4499 4917 2
4500 4918 2 LOCAL
4501 4919 2     PREV_LINE      : REF $BBLOCK;      ! Address of the previous line
4502 4920 2
4503 4921 2 IF .AED_W_TOTALSIZE GTR 0 OR .SEGMENT_SIZE GTR 0
4504 4922 2 THEN
4505 4923 2     BEGIN
4506 4924 2         PREV_LINE = .AED_L_LASTLINE;
4507 4925 2         TEMP_LINE = .AED_B_LINE;
4508 4926 2         UNTIL .PREV_LINE[LINE_V_BEGINACE]
4509 4927 2             OR .PREV_LINE[LINE_W_SIZE] GTR 0
4510 4928 2         DO
4511 4929 2             BEGIN
4512 4930 2                 PREV_LINE = .PREV_LINE[LINE_L_BLINK];
4513 4931 2                 TEMP_LINE = .TEMP_LINE - 1;
4514 4932 2             END;
4515 4933 2             IF .PREV_LINE[LINE_W_SIZE] EQL 0 THEN RETURN;
4516 4934 2             AED_COPSEGMENT (.PREV_LINE);
4517 4935 2             INSQUE (AED_T_CURLINE[LINE_L_FLINK], .PREV_LINE[LINE_L_BLINK]);
4518 4936 2             IF .AED_L_BEGINLINE EQL .PREV_LINE
4519 4937 2             THEN AED_C_BEGINLINE = AED_T_CURLINE[LINE_L_FLINK];
4520 4938 2             IF .AED_C_FIRSTLINE EQL .PREV_LINE
4521 4939 2             THEN AED_C_FIRSTLINE = AED_T_CURLINE[LINE_L_FLINK];
```

FINISH_ACE - Tie off the current ACE

```
4522 4940 3 IF .AED_L_LASTLINE EQL .PREV LINE
4523 4941 3 THEN AED_C_LASTLINE = AED_T_CURLINE[LINE_L_FLINK];
4524 4942 3 IF .INPUT_BUFFER[.SEGMENT_SIZE - 1] EQL ' '
4525 4943 3 THEN
4526 4944 4 BEGIN
4527 4945 4 AED_SET_CURSOR (.AED_B_LINE, .SEGMENT_SIZE);
4528 4946 4 SEGMENT_SIZE = .SEGMENT_SIZE - 1;
4529 4947 4 END;
4530 4948 3 IF .INPUT_BUFFER[.SEGMENT_SIZE - 1] NEQ 'C')
4531 4949 3 THEN
4532 4950 4 BEGIN
4533 4951 4 INPUT_BUFFER[.SEGMENT_SIZE] = ' ';
4534 4952 4 SEGMENT_SIZE = .SEGMENT_SIZE + 1;
4535 4953 4 END;
4536 4954 3 AED_L_LASTLINE[LINE_V_ENDACE] = 1;
4537 4955 3 NEW_TEXT_LINE = AED_REPSEGMENT ();
4538 4956 3 IF .TEMP_LINE GEQ 1
4539 4957 3 THEN
4540 4958 4 BEGIN
4541 4959 4 AED_POSITION (.NEW_TEXT_LINE);
4542 4960 4 AED_SET_CURSOR (.AED_B_LINE, .NEW_TEXT_LINE[LINE_W_SIZE]);
4543 4961 4 ECHO_DESC[DSCW_LENGTH] = 1;
4544 4962 4 ECHO_DESC[DSCSA_POINTER] = VECTOR [NEW_TEXT_LINE[LINE_T_TEXT],
4545 4963 4 .NEW_TEXT_LINE[LINE_O_SIZE] - 1; .BYTE];
4546 4964 4 AED_PUTOUTPUT (ECHO_DESC);
4547 4965 4 AED_POSITION (.AED_C_LASTLINE);
4548 4966 4 END;
4549 4967 3 END;
4550 4968 2 RETURN;
4551 4969 2
4552 4970 2
4553 4971 1 END;
```

! End of routine FINISH_ACE

001C 00000 FINISH_ACE:									
	54	0000'	CF	9E	00002	.WORD	Save R2,R3,R4	4884	
	53	0000'	CF	9E	00007	MOVAB	NEW TEXT LINE, R4		
		020C	C3	B5	0000C	MOVAB	SEGMENT SIZE, R3		
			04	12	00010	TSTW	AED_W_TOTALSIZE	4921	
			63	B5	00012	BNEQ	1\$		
			1F	13	00014	TSTW	SEGMENT_SIZE		
	52	8C	A3	D0	00016	BEQL	4\$		
F4	A4	FF6C	C3	9A	0001A	MOVL	AED_L_LASTLINE, PREV LINE	4924	
	0E	0A	A2	E8	00020	MOVZBL	AED_B_LINE, TEMP LINE	4925	
		08	A2	B5	00024	BLBS	10(PREV LINE), 3\$	4926	
			09	12	00027	TSTW	8(PREV LINE)	4927	
	52	04	A2	D0	00029	BNEQ	3\$		
		F4	A4	D7	0002D	MOVL	4(PREV LINE), PREV LINE	4930	
			EE	11	00030	DECL	TEMP LINE	4931	
		08	A2	B5	00032	BRB	2\$	4926	
			6A	13	00035	TSTW	8(PREV LINE)	4933	
			52	DD	00037	BEQL	10\$		
0000G	CF		01	FB	00039	PUSHL	PREV LINE	4934	
						CALLS	#1, AED_COPSEGMENT		

04	B2	F8	A3	0E	0003E	INSQUE	AED_T_CURLINE, @4(PREV_LINE)	4935
	52	90	A3	D1	00043	CMPL	AED_L_BEGINLINE, PREV_LINE	4936
			05	12	00047	BNEQ	5\$	
90	A3	F8	A3	9E	00049	MOVAB	AED_T_CURLINE, AED_L_BEGINLINE	4937
	52	88	A3	D1	0004E	CMPL	AED_L_FIRSTLINE, PREV_LINE	4938
			05	12	00052	BNEQ	6\$	
88	A3	F8	A3	9E	00054	MOVAB	AED_T_CURLINE, AED_L_FIRSTLINE	4939
	52	8C	A3	D1	00059	CMPL	AED_L_LASTLINE, PREV_LINE	4940
			05	12	0005D	BNEQ	7\$	
8C	A3	F8	A3	9E	0005F	MOVAB	AED_T_CURLINE, AED_L_LASTLINE	4941
	50		63	3C	00064	MOVZWL	SEGMENT_SIZE, R0	4942
	2B	0B	A340	91	00067	CMPL	INPUT_BUFFER-1[R0], #43	
			0F	12	0006C	BNEQ	8\$	
	7E		63	3C	0006E	MOVZWL	SEGMENT_SIZE, -(SP)	4945
	7E	FF6C	C3	9A	00071	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	00076	CALLS	#2, -AED_SET_CURSOR	
			63	B7	0007B	DECW	SEGMENT_SIZE	4946
	50		63	3C	0007D	MOVZWL	SEGMENT_SIZE, R0	4948
	29	0B	A340	91	00080	CMPL	INPUT_BUFFER-1[R0], #41	
			07	13	00085	BEQL	9\$	
0C	A340		29	90	00087	MOVB	#41, INPUT_BUFFER[R0]	4951
			63	B6	0008C	INCW	SEGMENT_SIZE	4952
	50	8C	A3	D0	0008E	MOVL	AED_L_LASTLINE, R0	4954
0A	A0		02	88	00092	BISB2	#2, -10(R0)	
0000G	CF		00	FB	00096	CALLS	#0, AED_REPSEGMENT	4955
	64		50	D0	0009B	MOVL	R0, NEW_TEXT_LINE	
		F4	A4	D5	0009E	TSTL	TEMP_LINE	4956
			39	15	000A1	BLEQ	11\$	
			64	DD	000A3	PUSHL	NEW_TEXT_LINE	4959
0000G	CF		01	FB	000A5	CALLS	#1, -AED_POSITION	
	50		64	D0	000AA	MOVL	NEW_TEXT_LINE, R0	4960
	7E	0B	A0	3C	000AD	MOVZWL	8(R0), -TSP	
	7E	FF6C	C3	9A	000B1	MOVZBL	AED_B_LINE, -(SP)	
0000G	CF		02	FB	000B6	CALLS	#2, -AED_SET_CURSOR	
EC	A4		01	B0	000BB	MOVW	#1, ECHO_DESC	4961
	51		64	D0	000BF	MOVL	NEW_TEXT_LINE, R1	4962
	50	0B	A1	3C	000C2	MOVZWL	8(RT), R0	4963
FO	A4	13	A140	9E	000C6	MOVAB	19(R1)[R0], ECHO_DESC+4	
		EC	A4	9F	000CC	PUSHAB	ECHO_DESC	4964
0000G	CF		01	FB	000CF	CALLS	#1, AED_PUTOUTPUT	
		8C	A3	DD	000D4	PUSHL	AED_L_LASTLINE	4965
0000G	CF		01	FB	000D7	CALLS	#1, -AED_POSITION	
			04	000DC	11\$:	RET		4971

: Routine Size: 221 bytes, Routine Base: \$CODE\$ + 357A

: 4554 4972 1
: 4555 4973 1 END
: 4556 4974 0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes								
AED_COMMON	1320	NOVEC,	WRT,	RD	NOEXE,NOSHR,	LCL,	REL,	OVR,NOPIC,ALIGN(0)		
\$OWNS	560	NOVEC,	WRT,	RD	NOEXE,NOSHR,	LCL,	REL,	CON,NOPIC,ALIGN(2)		
\$CODES	13911	NOVEC,NOWRT,		RD	EXE,NOSHR,	LCL,	REL,	CON,NOPIC,ALIGN(2)		
\$SPLITS	60	NOVEC,NOWRT,		RD	NOEXE,NOSHR,	LCL,	REL,	CON,NOPIC,ALIGN(2)		

Library Statistics

File	----- Symbols -----		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	54 0	1000	00:01.8
_\$255\$DUA28:[SYSLIB]TPAMAC.L32;1	42	0 0	14	00:00.1

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:AEDMAIN/OBJ=OBJ\$:AEDMAIN MSRC\$:AEDMAIN/UPDATE=(ENH\$:AEDMAIN)

Size: 13911 code + 1940 data bytes
Run Time: 03:15.4
Elapsed Time: 09:51.0
Lines/CPU Min: 1527
Lexemes/CPU-Min: 18458
Memory Used: 485 pages
Compilation Complete

0003 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

0004 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

REDMESSAG
LIS

REDPROMPT
LIS

SETACL
LIS

REDSUBR
LIS